ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

VISUAL LOOP CLOSURE DETECTION FOR AUTONOMOUS MOBILE ROBOT NAVIGATION VIA UNSUPERVISED LANDMARK EXTRACTION

M.Sc. THESIS

Evangelos SARIYANİDİ

Department of Control and Automation Engineering

Control and Automation Engineering Programme

JULY 2012

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL OF SCIENCE ENGINEERING AND TECHNOLOGY

VISUAL LOOP CLOSURE DETECTION FOR AUTONOMOUS MOBILE ROBOT NAVIGATION VIA UNSUPERVISED LANDMARK EXTRACTION

M.Sc. THESIS

Evangelos SARIYANİDİ
(504091106)

Department of Control and Automation Engineering

Control and Automation Engineering Programme

Thesis Advisor: Prof. Dr. Hakan TEMELTAŞ

JULY 2012

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

OTONOM MOBİL NAVİGASYON KAPSAMINDA ÇEVRİM KAPAMALARIN GÜDÜMSÜZ ÇIKARILAN GÖRSEL İMLEÇLER YARDIMIYLA SAPTANMASI

YÜKSEK LİSANS TEZİ

Evangelos SARIYANİDİ
(504091106)

Kontrol ve Otomasyon Mühendisliği Anabilim Dalı

Kontrol ve Otomasyon Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Hakan TEMELTAŞ

TEMMUZ 2012

Evangelos SARIYANİDİ, a **M.Sc.** student of ITU **Graduate School of Science Engineering and Technology** 504091106, successfully defended the **thesis** entitled "**VISUAL LOOP CLOSURE DETECTION FOR AUTONOMOUS MOBILE ROBOT NAVIGATION VIA UNSUPERVISED LANDMARK EXTRACTION**", which he prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor :	**Prof. Dr. Hakan TEMELTAŞ** İstanbul Technical University	.....
Jury Members :	**Prof. Dr. İbrahim EKSİN** İstanbul Technical University	.....
	Asst. Prof. Dr. Sanem SARIEL-TALAY İstanbul Technical University	.....

Date of Submission : 22 June 2012
Date of Defense : 17 July 2012

To my parents, brother and grandparents,

FOREWORD

First of all, I would like to express my gratitude to my supervisor Prof. Dr. Hakan Temeltaş, who has been a great influence on me and my interest in academic research, provided me with the opportunity to carry out research in the robotics laboratory for more than four years, and more importantly, given me a helping hand whenever needed. Thanks are due to my colleagues and friends at the robotics laboratory, especially Onur Şencan with whom I've been working for years. The study in this thesis has been partially supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) via the research project with grant number of 110E194.

Thanks are also due to Prof. Dr. Muhittin Gökmen, who has been extremely supportive of me in my graduate studies, and also has significant influence on my career and enthusiasm towards research. To my friends from the CVIP laboratory Birkan Tunç, Volkan Dağlı and Salih Cihan Tek for their support, great advices and valuable comments.

Finally, and most importantly, thanks are due to my family, who have been there through thick and thin, and supported me no matter what.

June 2012 Evangelos SARIYANİDİ

TABLE OF CONTENTS

Page

ABBREVIATIONS

BoW	**:** Bag-of-Words
CPU	**:** Central Processing Unit
ESS	**:** Efficient Subspace Search
FAB-MAP	**:** Fast Appearance Based Mapping
GPS	**:** Global Positioning System
GPU	**:** Graphical Processing Unit
LIDAR	**:** Light Detection And Ranging
PCA	**:** Principal Component Analysis
RAM	**:** Random Access Memory
ROI	**:** Region of Interest
SIFT	**:** Scale-Invariant Feature Transform
SLAM	**:** Simultaneous Localization and Mapping
SURF	**:** Speeded Up Robust Features

LIST OF TABLES

Page

LIST OF FIGURES

Page

VISUAL LOOP CLOSURE DETECTION
FOR AUTONOMOUS MOBILE ROBOT NAVIGATION
VIA UNSUPERVISED LANDMARK EXTRACTION

SUMMARY

Autonomous navigation is a very active research field in mobile robotics. Simultaneous localization and mapping (SLAM) is one of the major problems linked with autonomous navigation, which still remains as a challenging problem despite the extensive studies that have been carried out throughout the last decades. The SLAM problem becomes even more challenging when it is solved for large-scale outdoor environments.

One of the essential issues in SLAM is the detection of loop closures. Within the context of SLAM, loop closing can be defined as the correct identification of a previously visited location. Loop closure detection is a significant ability for a mobile robot, since successful loop closure detection leads to substantial improvement in the overall SLAM performance of the robot by means of resetting the most recent localization error and correcting the estimations over the past trajectory.

Vision based techniques have gained significant attention in the last decade, due mostly to the advances in computer processors and the development of certain effective computer vision techniques, which have been easily adapted to the loop closure detection problem. LIDAR has been used before the emergence of vision based techniques; however, it offered a limited capability for the solution of the loop closure detection problem.

In this thesis, a novel visual loop closing technique has been presented. The proposed technique relies on visual landmarks, which are extracted in an unsupervised manner. Image frames are represented sparsely through these landmarks, which are ultimately used to assess the similarity between two images and detect loop closing events.

Unsupervised extraction of visual landmarks is not a trivial task for several reasons. Firstly, a saliency criterion is needed to measure the saliency of a given image patch. Secondly, an efficient search algorithm is needed to test this certain saliency criterion all over an image and extract the most salient regions. In this thesis, the problem of extracting salient regions has been formulated as an optimization problem, where visual saliency has been described through an energy function and a Branch&Bound based search technique has been used to find the global maximum of this function. One of the contributions made in this thesis is the proposed saliency definition. An upper bound criterion, which facilitates efficient search through Branch&Bound, is the second contribution presented in this thesis.

The extraction of landmarks is the first step of the loop closing approach explained in this thesis. Once the landmarks are extracted, they are described and later re-identified using the well-established *ferns* classifiers. Place recognition, which ultimately leads

to loop closure detection, is achieved by means of a similarity function which measures the similarity between two images through the landmarks identified in each image.

The major difference among the method presented here and most of the methods that rely on local visual cues is that the local patches utilized in this study are specific to the environment they are extracted from. The results of the tests that have been performed on one of the most well-known outdoor datasets, indicate that the presented technique outperforms other well-known visual loop closure detection approaches.

OTONOM MOBİL NAVİGASYON KAPSAMINDA ÇEVRİM KAPAMALARIN GÜDÜMSÜZ ÇIKARILAN GÖRSEL İMLEÇLER YARDIMIYLA SAPTANMASI

ÖZET

Otonom navigasyon, mobil robotik alanında üzerinde en çok çalışılan konulardan biri olagelmiştir. Eş zamanlı Konum Belirleme ve Haritalama da (EZKH), otonom navigasyon konusu içinde en çok araştırılmış ve hala araştırılmakta olan problemlerden biridir. Ancak uzun soluklu çalışmalara rağmen, özellikle geniş ölçekli dış ortamlar baz alındığında EZKH kapsamında çözülmesi gereken birçok problem bugün hala mevcuttur.

EZKH bağlamında çevrim kapama problemi, otonom bir robotun daha önce bulunmuş olduğu bir yeri başarıyla tanıyabilmesi olarak özetlenebilir. Çevrim kapama çalışmalarının EZKH kapsamında ayrı bir önemi vardır, çünkü başarıyla gerçekleştirilen çevrim kapamalar robotun en güncel konumunu çok daha yüksek bir hassasiyetle belirleyip, geçmiş yörüngesindeki konumları üzerindeki kestirimlerini iyileştirmesine olanak sağlar. Konum kestirmede sağlanan bu iyileştirme, haritalama başarımını da önemli ölçüde artırır. Ancak öte yandan hatalı gerçekleştirilen çevrim kapamalar, EZKH kestirimlerindeki konum ve haritalama süreçlerinin hatalı biçimde güncellenmesine yol açacağı için, hatalı çevrim kapamaların genel EZKH sistemi üzerindeki etkisi yıkıcı boyutlara varabilir. Dolayısıyla hassasiyet, geliştirilen çevrim kapama sisteminde can alıcı bir öneme sahiptir.

Bir çevrim kapama sistemi tasarlanırken, dikkate alınması gereken kriterler yalnızca hassasiyet ve yüksek başarım değildir. En az bu iki kriter kadar önemli olan diğer bir kriter de sistemin hızı, ve dolayısıyla etkinliğidir. Bunun en önemli nedeni, EZKH sürecinin genellikle çevrimiçi bir süreç olması ve gerçek zamanlı işleyişin bir EZKH uygulamasında ayrı bir öneminin olmasıdır. Görüntü işleme tekniklerinin genel olarak yoğun işlem gerektiriyor olması da, etkin bir sistem tasarımını daha da güçleştirmektedir.

Çevrim kapama problemi, bu tez çalışmasında kamera algılayıcısı kullanılarak görüntü işleme teknikleriyle çözülmüştür. Görüntü işlemeye dayanan çevrim kapama problemi, temele indirgendiğinde bir görüntü eşleştirme, diğer bir deyişle görüntüler arasındaki benzerliği ölçme problemidir. Bu problem, birçok açıdan çözülmesi zor bir problemdir. Problemi zor kılan etmenler arasında en öne çıkanı, eşleştirilmeye aday görüntülerin çoğu durumda birbirine oldukça benziyor olmasıdır. EZKH probleminin dış ortamdaki olası uygulama alanları arasında çöl veya ormanlık alan gibi doğal ortamlar, veya sokak ve otoyol gibi kentsel ortamlar vardır. Bütün bu ortamlarda, birbirine benzeyen görüntülere sıklıkla rastlanabileceği için sistem kolayca yanılabilir. Bu durum, sistemin kolayca yanılmasına yol açabilir. Hatalı çevrim kapamaların genel EZKH sistemindeki yıkıcı etkisi gözönüne alınırsa, bu tip benzer görüntülerde yapılabilecek olası yanlış eşleştirmelere karşı özel bir önlem alınması gerekmekte olup,

çevrim kapama hipotezleri yeterince güvenilir olmadıkları sürece kesinlikle kabul edilmemelidir.

Bilgisayarla görüye dayanan teknilerin çevrim kapama probleminde kullanımı, son on yılda kaydedeğer ölçüde yaygınlaşmıştır. Bunun en önemli nedenlerinden biri, bilgisayar donanımı ve özellikle işlemci teknolojisindeki gelişmelerin, yoğun işlem gerektiren görüntü işleme yöntemlerinin kulanımını mümkün kılmasıdır. Diğer bir önemli etken de, çevrim kapama problemine uyarlanabilecek birçok bilgisayarla görü ve görüntü işleme tekniğinin önerilmiş olmasıdır. Kameradan önce kullanılan LIDAR gibi algılayıcılar, sözkonusu çevrim kapama problemini çözmekte kısıtlı olanaklar sunabilmişlerdir.

Bu tez çalışmasında, özgün bir çevrim kapama yöntemi sunulmaktadır. Önerilen yöntem, güdümsüz biçimde çıkarılan görsel imleçlere dayanmaktadır. Görüntüler imleçler yoluyla seyrek bir biçimde temsil edilmektir. Bu seyrek temsil yöntemi üzerinden görüntülerin eşleştirilmekte, ve en nihayetinde çevrim kapamalar saptanmaktadır.

Görüntüdeki çeşitli nirengi bölgelerinin güdümsüz bir biçimde saptanması için birtakım araçlar gerekmektedir. Öncelikle, verilen bir görüntü parçasının sıradışılığını ölçebilmek için bir matematiksel bir ölçüt bulunmalıdır. Bunun yanısıra, bu ölçütü görüntünün tüm alt bölgelerinde değerlendirip en sıradışı görüntü parçasının bulunmasında kullanılacak bir arama algoritması gerekmektedir. Bu tez çalışmasında, görsel imleç çıkarma problemi bir eniyileme problemi olarak düzenlenmiştir. Verilen bir görüntü parçasının sıradışılığını ölçmek için bir enerji fonksiyonu, arama yöntemi olarak da bir dal sınır arama yöntemi kullanılmıştır. Kullanılan enerji fonksiyonu bu çalışmadaki önerilen önemli yeniliklerden biridir. Ayrıca arama için kullanılan dal sınır yönteminin üst sınır kriteri de, önerilen enerji fonksiyonuna uyumlu olarak bu çalışma kapsamında önerilmiş diğer bir yeniliktir.

Görsel imleçlerin çıkarılması, çevrim kapama çalışmasının ilk adımını oluşturmaktadır. Çıkarılan imleçlerin tanımlanması, diğer bir deyişle daha sonra tekrar saptanabilmeleri için görünümlerinin öğrenilmesi gerekmektedir. İmleçlerin görünümlerinin öğrenilmesi ve saptanması için, bu konuda kabul görmüş önemli yöntemlerden olan *ferns* sınıflandırıcıları kullanılmıştır. Bu tekniğin kullanılmasındaki en önemli nedenlerden biri, sınıflandırıcı modelinin az sayıda imge ile eğitilebiliyor olmasıdır. İmleçlerin çevrim esnasında öğrenildiği gözönüne alındığında, bu özelliğin ne kadar önemli oldığı anlaşılabilir. Yöntemi öne çıkaran diğer bir niteliği ise, öğrenilen modelin yeni imgeler ışığında kolayca güncellenebilmesidir. Alışılagelmiş makine öğrenmesi tekniklerinden oldukça farklı olan bu teknik, bilinen yöntemler arasında probleme uygun olup kullanılabilecek tek yöntem olarak öne çıkmakta ve yüksek başarımla kullanılmaktadır.

Görsel imleçlerin çıkarılması ve öğrenilmesi ile, aracın yörüngesi üzerindeki yerler bu imleçler yardımıyla seyrek bir biçimde modellenmektedir. Bu şekilde modellenen yer imgeleri bir seyrek bir görünüm uzayı oluşturmaktadır. Görüntü eşleştirme ve çevrim kapama da bu uzayda gerçekleştirilmektedir. Yeni görüntüler, bu uzaydaki bütün yer imgeleriyle kıyaslanır en yakın eşleşme saptanır. Sözü geçen kıyaslama, bu tez kapsamında tanımlanan bir benzerlik fonksiyonuyla gerçekleştirilir.

Bir çevrim kapama hipotezinin başarıyla önerilebilmesi için, gelen görüntüye en benzer görüntünün doğrudan eşleştirilmesi yeterli değildir. Çevrim kapama hipotezinin oluşturulabilmesi için zorunlu bir koşul olarak, gelen görüntünün daha önce görüntülenmiş bir alanı temsil ettiği biliniyor olmalıdır. Dolayısıyla bir görüntünün daha önce görülüp görülmediğini ortaya çıkaracak bir yöntem gerekmektedir. Bu tez çalışmasında, gelen bir görüntünün daha önce görüntülenmiş bir alanı tesmil edip etmediğini ortaya çıkarmak için, görünüm uzayındaki en yakın eşleştirmenin etrafındaki yerel işaret değerlendirilmektedir. Bu işaret, bir görüntü daha önce gezilmiş bir alandan çıkarıldığında belirgin bir tepeye ve oldukça yüksek bir yerel maksimuma sahip olmaktadır. Öte yandan, bir görüntü daha önce görülmüş herhangi bir alandan çıkarılmamışsa, bu yerel işaret oldukça dağınık bir yapıdadır. Bu belirgin fark sayesinde, bir alanın daha önce görüntülenip görüntülenmediği kolayca anlaşılabilmektedir.

Görüntülerin eşleştirilmesi ve bu yolla çevrim kapama olaylarının saptanması ise, saptanan imleçleri girdi olarak alan bir benzerlik fonksiyonu kullanılarak gerçekleştirilmektedir. Vektör normları üzerinden tanımlanan bu benzerlik fonksiyonu, basit ve anlaşılır bir yapıda olmakla beraber yüksek başarımlı benzerlik sonuçları üretmektedir.

Bu tezde sunulan çalışmanın bilimsel yazındaki diğer yerel görsel imleçlere dayanan yöntemlerle arasındaki en temel ayrım, imleçlerin robotun gezdiği ortamlardan çıkarılmasıdır. Diğer çalışmalardaki genel yaklaşım, belirli imleçlerin geniş veritabanlarından çıkarılıp öğrenilenmesi yönündedir. Bu çalışmada önerilen yöntem, bilinen diğer görsel çevrim kapama yöntemleriyle en kabul görmüş veritabanlarından biri üzerinde karşılaştırılmıştır. Elde edilen sonuçlar, çalışmadaki yaklaşımın ve genel olarak önerilen yöntemin bilinen diğer yöntemlerden daha üstün olduğunu göstermektedir.

1. INTRODUCTION

Autonomous navigation has been, and still is a very attractive research field of mobile robotics. The SLAM problem is one of the major problems linked with autonomous navigation, and despite the extensive studies that have been carried out for years, there is still considerable room for improvement.

1.1 Problem Statement

Loop closure detection, one of the most prominent subproblems of the general SLAM problem, can be defined as the correct identification of a previously visited location. Loop closure detection is an extremely significant ability for a mobile robot which performs SLAM, since correct loop closure detections augment both the localization and mapping processes.

The self-location estimations obtained from the SLAM process are always erroneous, and even the slightest errors are accumulated up to the point that they can't be dealt with. The most straightforward way to cope with the accumulated localization errors, is to occasionally reset them by closing loops. Successfully detected loop closing events, provide a more precise estimation over the self-location of the robot, by associating the current location with a location from the past trajectory, which is associated with a more accurate location estimation. Closing loops has also a positive effect on the past trajectory of the robot, since all of the estimations over the past trajectory are updated and corrected. Localization and mapping are tightly coupled processes; therefore, any corrections made on the self-location estimations, immediately improve the accuracy of the mapping process. It is obvious that correctly closed loops, have a significant effect on the overall SLAM procedure.

Loop closure detection however is a double-sided sword. Even though correctly detected loop closures improve the SLAM performance, flawed loop closures have an extremely adverse effect on it — false loop closure detections cause the entire

trajectory to be updated with incorrect data, which is catastrophic for both localization and mapping processes. Therefore, it is vital that the loop closure detection system is extremely accurate and precise; therefore, loop closure hypotheses shall not be accepted unless they are highly reliable.

High accuracy is not the single criterion that must be considered when designing a loop closing system. SLAM applications are usually on-line processes, hence the loop closing system in question must be operating very fast. This restriction makes the system design even more challenging for two reasons. Firstly, image processing techniques are computationally heavy, especially when the whole incoming image is being processed; therefore, the effort spent to process each single frame must be minimized. Secondly, the descriptor vector of each incoming image must be compared with all previously extracted image descriptors, and this comparison will not allow real-time operation if the dimensionality of the search space is high and the trajectory that is being planned to traverse is long. In other words, the loop closing system that is being designed must be spending very little effort processing each image, and the descriptor for each image must be small in dimension if on-line operation is desired.

A major issue that must be dealt with is perceptual aliasing, which occurs when certain places look very similar due to their nature, *e.g.* forests, railroads, office corridors etc. Triggering false alarms is very likely when perceptual aliasing is present; therefore, perceptual aliasing must be carefully considered in the system design.

On the other hand, a common opinion of many researchers dealing with loop closure detection is that the data used to develop loop closure detection hypotheses must be independent from the estimations and outcome of the SLAM process [1–3], *e.g.* map feature positions or vehicle location/speed, since these estimations are erroneous and aimed to be corrected. In other words, dedicated loop closing mechanisms that are fed from sources independent from the SLAM process are more reliable than the ones utilizing the SLAM outcome.

Using cameras to achieve loop closure detection has become feasible and extremely popular in the last decade, and unsurprisingly, most notable techniques to date rely on visual sensory. The data provided by camera is more rich and detailed than the data

provided by sensors like LIDAR. However, using cameras has certain shortcomings that must be addressed. The most eminent issue is the sensitivity against illumination, which is not in question when other sensors like LIDAR are used. Illumination conditions are subject to change very often; therefore, any visual loop closure system must be robust against illumination up to a certain point. The sensitivity to view perspective is also another concern that must be pointed out and dealt with. There are also issues like robustness against scaling, rotation or translation, however, these are issues that are common for most kinds of sensors.

In summary, loop closure detection is an active and challenging problem that must be handled in real-world SLAM applications. Any solution to this problem must be very accurate and computationally efficient. Furthermore, it must be independent from the outcome of the SLAM process and moreover, perceptual aliasing must be considered. In this thesis, a novel visual loop closure detection system, which considers all of these issues has been proposed. The literature review has been presented in the next section, and the approach proposed in this thesis has been summarized in the subsequent section.

1.2 Literature Review

The importance of loop closure detection for Simultaneous Localization and Mapping (SLAM) algorithms has been established by many authors in numerous studies [2–9]. Various approaches have been proposed to solve this problem. On the other hand, the significance of using dedicated mechanisms for detecting loop closures has been highlighted by several authors [2, 4].

In [7], Williams *et al.* present a comparison on visual loop closure techniques that rely on monocular vision. According to this comparison, vision based loop closing techniques come in three broad categories: Map-to-map techniques, image-to-map techniques and image-to-image techniques. The map in this context involves the maps produced as part of the mapping of the overall SLAM process. It is obvious that the comparison in [7] is made according the the information that is used to close loops. Dedicated visual loop closure techniques, which are the techniques that don't utilize the estimations of the SLAM process, fall into the category of image-to-image

techniques. The study that is being presented in this thesis falls into this category, and the emphasis is put on the methods falling into this category on rest of this section.

Early studies on visual loop closure were aimed at describing each image with a single descriptor vector extracted from the whole scene. These kind of descriptors are usually referred to as global descriptors. Basically, there are two ways to extract global descriptors from images. 1) Using image processing/analysis techniques and extract descriptors out of texture transformations, histograms, edge information etc. 2) Using dimensionality reduction techniques and represent images in a lower-dimensioned space.

There have been proposed several techniques that aimed at place recognition using global image descriptors. Ulrich and Nourbakhsh used a set of image histograms to extract global descriptors out of images [10]. Lamon *et al.* used features extracted from color and edge information [11]. Torralba *et al.* represented images with a set of features extracted out of texture information [12].

Many researchers have adopted existing or developed new dimensionality reduction techniques to achieve loop closure detection. Kröse *et al.* have used PCA to represent images and search for loop closure detection in a lower dimensional space [13]. Another approach that relies on dimension reduction to extract global descriptors has been proposed by Ramos *et al.*, where a dimensionality reduction technique has been combined with variational Bayes learning to extract a generative model for each place [14]. Bowling *et al.* utilize an unsupervised approach in [15], which uses a sophisticated dimensionality reduction technique in order to extract descriptors for images.

Visual loop closure detection systems that rely on global descriptors however, are quite fragile, since the appearance of an entire image is very sensitive to illumination and view perspective changes. The usage of local descriptors for several recognition tasks has been very popular in the computer vision community. The striking study of Lowe [16], which introduces the SIFT features, has proven that local descriptors are much more robust against illumination and view perspective changes. SIFT features have been used very widely for numerous recognition tasks, including place recognition.

The major downside of these features is that their extraction is computationally intensive, which makes their real-time operation infeasible. Many similar studies have been carried out, and to date, the SURF features proposed by Bay *et al.* in [17] are among the most popular key point descriptors, due mostly to the balance between their computational complexity and their robustness. Another groundbreaking study is the Bag-of-Words (BoW) model proposed by [18], which also had many applications. The BoW model has also had several applications in the robotics field. This model, is based on building a visual vocabulary by clustering key point descriptors extracted through a large dataset. The clustered descriptors are referred to as visual words, and its a common practice to compute the empirical appearance probabilities of these words in order to develop a probabilistic recognition framework.

Local visual features, which prove to be very effective, have been frequently used by the robotics community for several tasks. Newman and Ho are among the first ones to suggest the advantage of using certain salient features rather than features extracted out of the entire image in [4]. Another early study is the one of Li and Kosecka [19], which also concentrates on finding the most salient regions in images. Wang *et al.* use a visual vocabulary, which is constructed in an off-line fashion, and use this vocabulary to extract descriptors based on the BoW model. On the other hand, Filiat *et al.* do similarly utilize a BoW model, which relies on a visual vocabulary that is built on-line. In [20], Ferreira *et al.* similarly employ a BoW model where they consider learning the dependency between the visual words using Bernoulli mixtures. Other techniques that use local visual cues are [6, 21, 22].

The groundbreaking FAB-MAP technique proposed by Cummins and Newman [3], utilizes the BoW model in a generative probabilistic framework. In the proposed study, Cummins and Newman use the BoW model constructed out of SURF features in a generative probabilistic framework. A generative model is constructed for each location. This probabilistic model considers the statistical dependencies among visual words up to the second degree via Chow-Liu approximation [23], in order to cope with the perceptual aliasing problem. Moreover, Monte-Carlo sampling is employed in order to reveal whether a location has been visited before or not. The performance of the FAB-MAP technique is impressively high — even more researchers have moved

towards using local visual cues to achieve loop closure detection after the impressive results of this study.

The local techniques listed so far do mostly utilize very small, low-level key point descriptors, and use them in conjunction with a BoW model to learn the visual words in an off-line fashion. The fact of the matter is that the visual words in this context are generic words. In contrast to this point of view, this thesis introduces a loop closure detection framework that utilizes visual landmarks that are specific to the environment that they are being extracted from. Moreover, these landmarks are relatively larger patches varying in size, unlike the small key point descriptors whose size is fixed. The study of Espinace *et al.*, similarly considers the extraction of visual landmarks out of the environment that the vehicle is navigating.

The technique that has been presented in this thesis has been developed by considering the outcome of several visual loop closure detection techniques. It is obvious that, using local features is very beneficial for several reasons. However, in contrast to most studies, the study that has been carried out in this thesis focuses on extracting landmarks specific to the environment that the robot traverses. The motivation behind this point of view is that humans and many living beings do successfully use visual landmarks for navigation [24–26]. The technique that has been developed in this thesis has been summarized in the following section.

1.3 Hypothesis

The technique presented in this thesis is motivated by the success of the visual loop closure techniques that utilize local features, and the fact that most animals successfully use visual landmarks for navigation and place recognition. This technique relies on unsupervised landmark extraction to achieve place recognition and ultimately loop closure detection.

Loop closure via unsupervised landmark extraction involves three major components: 1) Finding salient regions in images to use as visual landmarks, 2) learning the appearance of the extracted landmarks to describe and re-identify them, 3) matching images which are sparsely represented through the identified landmarks.

Within the scope of this thesis, the problem of unsupervised landmark extraction has been formulated in an optimization framework, where the objective function describes the saliency of a given image patch. This objective function is an energy function and a Branch& Bound based search technique has been employed to find the global optimum of it. This landmark extraction scheme is the major contribution of this thesis. The proposed energy function considers for saliency twofold: 1) Saliency among frames, 2) saliency within a single frame. It not only provides a different point of view to the problem saliency detection, but also operates very efficiently when combined with the proposed Branch&Bound search technique. This Branch&Bound technique is basically based on the study of Lampert *et al.* in [27] — this is a basic yet effective image search framework, which requires an upper bound criterion compatible with the objective function. In other words, it is a generic technique that needs a specific upper bound criterion, which must be compatible with the objective function. This upper bound has also been defined in this study, and speed performance results indicate that it enables very efficient search.

There are various out-of-box classifiers that may be used to learn the appearance of the extracted landmarks. However, there are certain restrictions that narrow the choices down to a few: The number of positive samples which can be used to learn the appearance of the landmarks is quite limited in this case; therefore, it is crucial that the classifier can generalize with very few samples. Moreover, the technique in question must be very efficient both in training and testing phases. The well-established *ferns* classifier has been utilized, since it satisfies these requirements and performs quite well.

A landmark database is used to save the landmarks' statistics. This database is initially empty, and it is updated on-line throughout the trajectory. The detection statistics of each landmark are saved to this database — these statistics are used to assign an empirical detection probability for each landmark and use this probability to describe the distinctiveness of each landmark.

According to the technique described in this thesis, incoming frames are represented sparsely through landmarks whose appearance has already been learnt *on-the-fly*. The next step to accomplish is comparing images through their sparse representation in

order to find the best matches and cast a loop closure hypothesis. In this thesis, a similarity function, which considers the detection confidence and spatial location of each landmark is employed for this purpose.

The proposed loop closure detection technique has been evaluated on two datasets: 1) The new college dataset [28], an outdoor dataset collected with a panoramic camera mounted on the top of a wheeled mobile robot, 2) The ITU Robotics Laboratory indoor dataset, an indoor dataset collected with a hand-held camera. Results indicate that the proposed loop closure detection framework performs with high accuracy, and outperforms the techniques known to date.

There are two publications involved with this thesis: The first paper describes the landmark extraction process: [29], and the second [30] puts emphasis on the overall loop closure detection framework.

2. UNSUPERVISED VISUAL LANDMARK EXTRACTION

The term saliency does not have a clear definition; in this study it has been used to describe certain pre-attentively distinctive image patches, which are suitable to represent place images in a sparse manner. Extracting regions with a semantic meaning is not strictly expected, yet it occasionally occurs. This chapter focuses on explaining how the saliency of a given image patch has been measured. As it has been stressed earlier, this has been accomplished through an energy function, which is actually the objective function of the optimization framework that has been proposed in order to extract visual landmarks.

2.1 Visual Saliency Definition

The optimization framework that has been used for visual landmark extraction, operates on an alternative image representation. The saliency is defined over the features of this representation, where the search to find the optimum output is also is also being performed. In other words, the intensity image is transformed into another plane before the landmarks are extracted from it.

According to this representation, an image I is composed of N features which are denoted with $F_I = \{f_1, \ldots, f_N\} \subset \mathcal{F}$, and it is assumed that the marginal probability of observing each of those features $p(f_i)$ is known. Furthermore, an arbitrary rectangular region within I has been shown with Ω. The number of features falling into the region Ω, has been given with the simple function $K(\Omega)$. Moreover, F_Ω has been used to denote the set of features lying inside Ω, so that $F_\Omega = \{f_{\omega_1}, \ldots, f_{\omega_{K(\Omega)}}\}$. The representation that has been described so far, has been illustrated in Figure 2.1.

The probability of observing the region Ω can be expressed as the joint probability $P(F_\Omega)$. Under these assumptions, the problem of finding a distinguishing rectangular patch Ω^* inside an image I, can be converted to the problem of finding the feature set F_{Ω^*} with the lowest joint probability. However, since $F_{\Omega^*} \subset F_I$, F_{Ω^*} is bound to be

F_I. In other words the largest feature set is the most distinguishing combination inside an image.

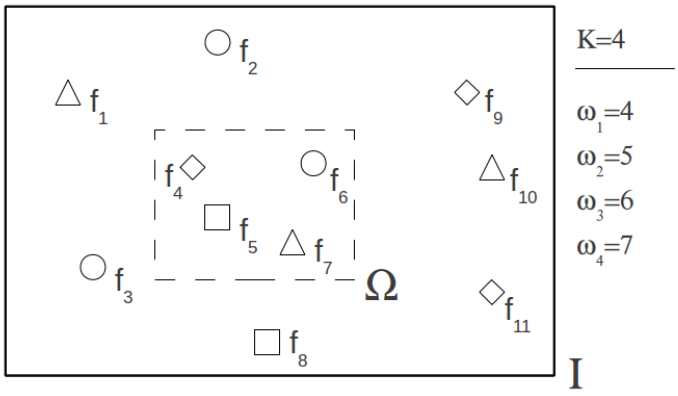

Figure 2.1: An illustration of the image features and a sample rectangular region.

One way to find a smaller and denser salient patch is formulating the salient patch detection problem as an energy maximization problem and enforcing a size constraint on the energy function. Let $H(\Omega)$ be a function which gives the area of a given rectangle. The energy function with the size constraint is:

$$E(F_\Omega, \Omega) = -P(F_\Omega) + \lambda_1 H(\Omega), \tag{2.1}$$

where $\lambda_1 \leq 0$. Due to this criterion, the size of the output region can be tuned by the constant λ_1.

This function may be modified to meet several needs by enforcing additional constraints such as:

- A constraint on the quantity of features to limit the number of features lying inside the rectangle,
- A constraint on the $3D$ depth of the features to enforce them to be coplanar, if such a depth information exists *e.g.* if a stereo imaging device is being used.

In the case that the feature quantity constraint is enforced, the energy function is reformulated as follows:

$$E(F_\Omega, \Omega) = -P(F_\Omega) + \lambda_1 H(\Omega) + \lambda_2 K(\Omega). \tag{2.2}$$

where $\lambda_2 \leq 0$ if the number falling into the output region is intended to be restricted.

Basically, the saliency criterion adopted in this thesis, relies on the terms present in (**2.2**). The constraints on (**2.2**) are enforced to satisfy two heuristics: 1) Small and dense salient regions are more notable than large ones, 2) saliency that is achieved with few features is more valuable.

On the application side, the features are assumed to be statistically independent to make the computation of the joint probability term $P(F_\Omega)$ tractable. However, a more complex statistical model and/or inference may always be employed if possible. The energy function in (**2.2**) is reformulated under the Naive-Bayes approximation such as:

$$E(F_\Omega, \Omega) \approx P(f_{\omega_1}) \dots P(f_{\omega_{K(\Omega)}}) + \lambda_1 H(\Omega) + \lambda_2 K(\Omega), \tag{**2.3**}$$

and the salient regions are detected by maximizing this function:

$$\Omega^* = \underset{\Omega}{\operatorname{argmax}}\, E(F_\Omega, \Omega). \tag{**2.4**}$$

The features f_i used in this work are the visual words of the BoW model [18,31] which have been successfully used in many studies including studies related to vision-based loop closure detection. The statistics of the visual words are computed off-line through a very large dataset. In this case, the visual vocabulary has been formed by clustered SURF features [17] and the statistics of the words are kept inside this vocabulary — the aforementioned marginal probabilities $p(f_i)$ are nothing more than the empirical probabilities inferred out of this vocabulary. The vocabulary that has been used in this study is the one presented by Cummins and Newman in [3].

Using a SURF based BoW model is not the only option to put the energy minimization scheme described so far into work, however, it has been used as an easy to implement out-of-box solution which has proven to be efficient. The truth of the matter is it that using the proposed BoW model is probably an over complex, computationally intensive solution. As illustrated in Chapter 6, it is eminent that the bottleneck of the overall loop closure detection framework is the computation of the SURF features. The SURF model, and even the entire BoW model can be replaced with an alternative model, as soon as the alternative model offers marginal probabilities assigned to its

features. In other words, the optimization framework has been described in a generic fashion, and any feature set consisting of features f_i can be used, under the assumption that the probability of the features $p(f_i)$ can be inferred.

2.2 Dealing with Perceptual Aliasing

One of the most challenging issues in place recognition is perceptual aliasing — environments where repetitive structure is present. Any technique that aims to achieve loop closure detection must account for perceptual aliasing, since perceptual aliasing is present in many indoor and outdoor environments: Offices, forests, railroads etc. This fact must be considered when the landmarks to represent the locations are being extracted.

When perceptual aliasing is present, certain features f_i will be extracted multiple times from certain scenes; *e.g.* if f_i somehow describes a leaf of a tree image, it will be extracted multiple times in any forest image. In order to deal with perceptual aliasing, the weight of each feature has been adjusted in proportion with the number of appearances of that feature inside the subject image. To achieve this, the well-established *tf-idf* (term frequency - inverse document frequency) score [32] has been used. In few words, this score accounts both for the frequency of a word within a single image through the *term frequency* score (*tf*) and for the frequency of the word inside the large visual dataset through the *inverse document frequency* (*idf*).

The *tf-idf* score has been introduced through modifying (**2.3**) by replacing the individual probability terms with the *tf-idf* score of each feature:

$$E(F_\Omega, \Omega) \approx \text{tf-idf}(f_{\omega_1}, I) \dots \text{tf-idf}(f_{\omega_{K(\Omega)}}, I) + \lambda_1 H(\Omega) + \lambda_2 K(\Omega). \qquad \textbf{(2.5)}$$

The salient patches can be considered as salient only in the context of the image. For instance, a tree is not a salient visual patch in a forest image; however, it might be discriminative in an urban scene. On the other hand, a traffic sign might not be a salient patch in an urban scene, whereas it might turn out significant in a natural scene. The *tf-idf* score enables such a discrimination and deals with the perceptual aliasing

problem by lowering the weight of the words that appear frequently inside the same image.

It is worthwhile to note that in Section 2.3, an objective function, which in this case is the proposed energy function, is needed to derive an upper bound condition for the Branch&Bound based search [27]. The formula with probability terms in (**2.3**) has been used rather than (**2.5**) since (**2.3**) is a simpler equation and its more intuitive. However the upper bound of the technique can be derived (and actually is being derived) from (**2.5**).

Exemplar output, pointing to the optimum of (**2.5**) is shown in Figure 2.2 — the salient regions on these images are extracted by setting λ_1 to $\lambda_1 = 0.015$ and λ_2 to $\lambda_2 = 0$. As it has been stressed earlier, output with a semantic meaning may be extracted through this function, even though it is not strictly expected. Certain output in Figure 2.2 point to regions with semantic meaning like plates, vehicles etc. An exemplar failure case of the algorithm has also been depicted with the last image of the last row. The central and right images on the second row, illustrate the consistency of the energy function.

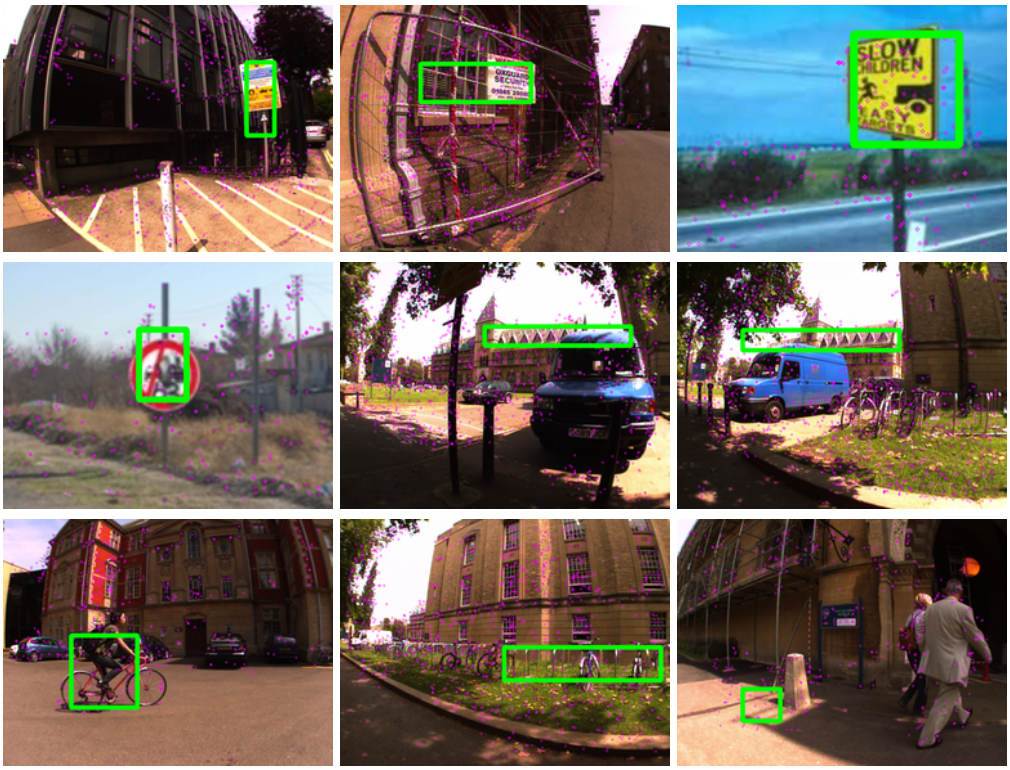

Figure 2.2: Exemplar salient patches.

At this point, it is worth to stress that the actual landmarks that are used to represent locations and cast loop closure estimations, are not similar to the ones shown in Figure 2.2. The constraint parameters $\lambda_{1,2}$, have been adjusted in a way to output smaller landmarks. Furthermore, multiple landmarks have been used to describe scenes, rather than a single landmark for each scene. Some exemplar landmarks that have actually been used to represent scenes have been shown on Figure 2.3. The salient regions in these images are extracted with the parameters $\lambda_1 = 0.020$ and $\lambda_2 = 10^{-6}$. The number of landmarks used to describe scenes and the way that multiple landmarks are extracted has been explained in Section 4.1.

Figure 2.3: Exemplar salient regions used to represent locations.

2.3 Searching the Most Salient Region: Branch&Bound Optimization

In order to find the most salient region inside an image, a search method is needed to test the criterion given in (**2.3**) all over the image. An extremely efficient search technique has been employed to find the optimum of this objective function.

Employing a brute-force search to perform the maximization in (**2.4**) is not an option due to numerous candidate windows. Using the well-known sliding window technique [33], is also not suitable — this technique has also a large computational cost and it also requires prior knowledge about the width/height ratio of the output rectangle.

2.3.1 Why Branch&Bound optimization?

The number of different image search techniques developed and used by the computer vision community is quite limited to date. One of the most eminent reasons for the shortage of efficient image search techniques is that unlike a regular optimization problem, there is no continuity on the function to optimize in most problems – image

search techniques are mostly needed for object detection, and the actual position of the object is completely random, output of the neighbouring windows don't give a clue about the existence of the object in search. Almost all object detection methods make use of a classifier which can classify whether a certain image patch completely contains the object and only the object. This formulation barely enables the usage of different optimization techniques. This is the reason that most vision based detection methods rely on brute-force search techniques like the *sliding windows* [33] technique.

However, the search task in this study is different from a regular object search. There is an objective function to be optimized, and the output of each candidate window is in correlation with its neighbours. This framework allows for a more efficient search technique, however the options are still quite limited, since very few efficient search tehcniques have been proposed by the computer vision community, and even fewer of the ones proposed are in harmony with the objective function in (**2.3**).

Fortunately, an image search technique that uses Branch&Bound search has been developed by Lampert et al. [27], which aims to be an efficient alternative to the basically brute-force *sliding windows* technique. This Branch&Bound based technique is referred to as Efficient Subwindow Search (ESS), and it actually is a generic image search technique — it can't be directly applied for any problem. It requires the definition of a proper upper bound criterion. The technique has been explained in details in Section 2.3.2.

It's worthwhile to define the Branch&Bound search and describe its search procedure in few words, before proceeding with its application on this thesis. Branch&Bound is a general method for finding optimal solutions of discrete and combinatorial optimization problems. These problems are easy to state and they have a finite solution. However, the number of all feasible solutions is usually very large; therefore, finding the optimal solution might require a great computational effort.

Let's assume that the problem in question is a minimization problem. The key idea of Branch&Bound is finding a certain output value $\hat{y}_j$ that would speak for a set of candidate solutions $\mathcal{X}_j$, so that the output of any solution from $\mathcal{X}_j$ would never be lower than $\hat{y}_j$. Once a single solution $x_t \notin \mathcal{X}_j$ which with a value y_t so that $y_t < \hat{y}_j$ is

found, then the whole set $\mathcal{X}_j$ is safely discarded, knowing that it would never contain the minimum solution. The crucial point in this procedure is having an upper bound criterion, which would be used to compute the upper bound of a given candiate set $\mathcal{X}_j$. The upper bound value that is calculated for a set $\mathcal{X}_j$ through this criterion should never be lower than any of the solutions in $\mathcal{X}_j$ — it should not violate the upper bound condition. Moreover, the upper bound should also not held extremely high to ensure that the upper bound condition is not violated, since if it is extremely high, it would be hard to discard sets by finding certain y_t so that $y_t < \hat{y}_j$.

The Branch&Bound based image search technique has ben described in Section 2.3.2, and the upper bound criterion used in conjunction with this technique has been defined in Section 2.3.3.

2.3.2 Efficient Subwindow Search

An efficient Branch&Bound based image search technique has been proposed by Lampert *et al.* in [27]. This technique is referred to as *Efficient Subwindow Search* (ESS), and it is a generic image search technique, which needs to be adapted to the application that it is being applied to, by defining an upper bound compatible with the objective function. This technique is especially attractive for two reasons: 1) It finds the global optimum of the given function; 2) it facilitates very efficient search by discarding most of unpromising regions.

As it has been stressed earlier, this search method requires an efficient upper bound criterion to operate efficiently. This criterion is used to compute the highest possible response of any rectangle lying inside a given rectangle set. Following the notation in [27], a rectangle is parametrized by its top, bottom, left and right coordinates (t, b, l, r). Furthermore, a rectangle set is defined as any rectangle of which the coordinates remain in predefined intervals, which are represented as $[T, B, L, R]$ where $T = [t_{low}, t_{high}]$ etc. This representation has been illustrated in Figure 2.4.

The operation of the generic ESS algorithm on an image I of size $n \times m$ is as follows. The algorithm requires the upper bound $\hat{E}$, which has been described earlier in this section and is given in (**2.10**). An initially empty priority queue depicted with P is constructed. The algorithm begins by computing the upper bound of the largest

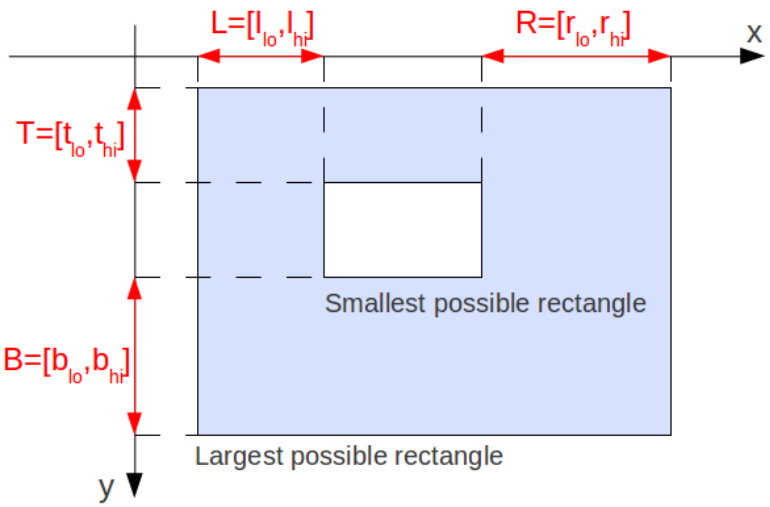

Figure 2.4: An illustration for the rectangle parametrization of ESS.

possible rectangle set in image $\mathbf{\Omega} = [[1,n],[1,n],[1,m],[1,m]]$, and adding to P, where the sets are listed according to their upper bounds — the set with the largest upper bound is placed on the top of P. Then, this set is continuously split to disjoint child sets. The upper bound of each set is calculated and each set is added to P according to its upper bound value. Once children sets are pushed to P, the parent set is removed from it. The algorithm continues this splitting procedure by beginning with the set on the top of P in order to process promising sets first. The strategy of proceeding by beginning with the most promising candidates, is the general rule of the Branch&Bound search theory [34]. It has been shown that giving precedence to the most promising candidates improves the speed of the search algorithm significantly. The pseudo-code of the ESS has been given in the following Algorithm.

The details of the ESS algorithm that have not been given here can be found in [27]. The following section, describes the upper bound condition which enables the usage of ESS for the visual landmark extraction framework described in this thesis.

2.3.3 Definition of the upper bound criterion

A valid upper bound for a rectangle set $[T, B, L, R]$ must hold for all the rectangles inside this set — the output of any rectangle in this set can not be larger than this bound. On the other hand, the efficiency of the method may be decreased if the bound

Algorithm 1 Efficient Subwindow Search through Branch&Bound

Require: image I
Require: upper bound function $\hat{E}$
Ensure: $\Omega^* = \text{argmax}_{\Omega \in \mathbf{\Omega}} E(\Omega)$
  initialize P as empty priority queue
  set $[T, B, L, R] = [1, n] \times [1, n] \times [1, m] \times [1, m]$
  repeat
    split $[T, B, L, R] \rightarrow [T_1, B_1, L_1, R_1] \dot{\cup} [T_2, B_2, L_2, R_2]$
    push $[T_1, B_1, L_1, R_1]$; $\hat{E}([T_1, B_1, L_1, R_1])$ onto P
    push $[T_2, B_2, L_2, R_2]$; $\hat{E}([T_2, B_2, L_2, R_2])$ onto P
    retrieve top state $[T, B, L, R]$ from P
  until $[T, B, L, R]$ consists only of one rectangle
  set $\Omega^* = [T, B, L, R]$

is held extremely high to ensure this condition. A compatible upper bound criterion has been proposed in the scope of this thesis. This criterion is high enough to ensure that the bound upper bound condition is not violated and low enough to facilitate efficient search. Let $\Omega_\cup$ be the largest possible, $\Omega_\cap$ the smallest possible rectangle and Ω any arbitrary rectangle in a rectangle set $\mathbf{\Omega} = [T, B, L, R]$. The following inequalities hold for all $\Omega \in \mathbf{\Omega}$ (recall that $\lambda_1, \lambda_2 \leq 0$):

$$-P(F_{\Omega_\cup}) \geq -P(F_\Omega) \tag{2.6}$$

$$\lambda_1 H(\Omega_\cap) \geq \lambda_1 H(\Omega) \tag{2.7}$$

$$\lambda_2 K(\Omega_\cap) \geq \lambda_2 K(\Omega). \tag{2.8}$$

If the above-written inequalities are summed, the following inequality is obtained:

$$-P(F_{\Omega_\cup}) + \lambda_1 H(\Omega_\cap) + \lambda_2 K(\Omega_\cap) \geq -P(F_\Omega) + \lambda_1 H(\Omega) + \lambda_2 K(\Omega). \tag{2.9}$$

The left side of (**2.9**) can be interpreted as an upper bound over a rectangle set Ω, which is denoted with $\hat{E}(F_{\mathbf{\Omega}}, \mathbf{\Omega})$ and expressed in terms of the smallest and the largest rectangle contained in this set:

$$\hat{E}(F_{\mathbf{\Omega}}, \mathbf{\Omega}) = -P(F_{\Omega_\cup}) + \lambda_1 H(\Omega_\cap) + \lambda_2 K(\Omega_\cap). \tag{2.10}$$

Suppose that there is a rectangle set $\mathbf{\Omega}_i$ inside an image with an upper bound $\hat{E}(F_{\mathbf{\Omega}_i}, \mathbf{\Omega}_i)$, and a single rectangle Ω_α inside the same image such as $\Omega_\alpha \notin \mathbf{\Omega}_i$ and its response to the energy function is $E(F_{\Omega_\alpha}, \Omega_\alpha)$. If $E(F_{\Omega_\alpha}, \Omega_\alpha) > \hat{E}(F_{\mathbf{\Omega}_i}, \mathbf{\Omega}_i)$, then

it is ensured that any rectangle in $\mathbf{\Omega}_i$ can not be the global optimum of the function in (**2.2**). Hence, the whole rectangle set is discarded safely.

2.4 Efficient Implementation via Integral Images

Although the search technique in question is quite fast, there is still room for substantial improvement on the computational effort. The computation of the joint probability of the features F_Ω inside a rectangular region Ω, requires $K(\Omega)$ computations under the naive-Bayesian assumption as it is seen in (**2.3**). The computational overhead might increase dramatically if the number of features falling inside the region is excessive. In order to deal with the potential problems that may arise from excessive features, the integral images [35, 36] have been employed. Thanks to integral images, the computation of the joint probability of a given region is achieved quite efficiently with only four computations. In this section, the usage of integral images in the context of the proposed optimization framework has been described.

In order to make the computations tractable, a sparse image representation that is inspired from the illustration in Figure 2.1 is being utilized. The adopted representation is illustrated in Figure 2.5.

First the visual words f_i are extracted out of intensity images. Then, a new, denser image that is composed only of these visual words is formed – the rows and columns that don't contain visual word are discarded. This new image, which is a much smaller and denser image, is denoted with I_F, and it is shown symbolically in the middle step of the process shown in Figure 2.5. This image is simply constructed as follows:

$$I_F(x_i, y_i) = \begin{cases} \ln p(f_{(i,j)}) & \text{if there is a visual word in image coordinate } (i,j) \\ 0 & \text{otherwise.} \end{cases} \quad \textbf{(2.11)}$$

As it is seen in (**2.11**), I_F contains the marginal appearance probabilities $p(f_i)$ of the visual words (See Section 2.1). Once the image I_F is formed, the next step is to construct the integral image of this image. The integral image, denoted with II_F, is created through such as:

$$II_F(x_i, y_j) = \sum_{k=1}^{i} \sum_{p=1}^{j} I_F(x_k, y_p). \quad \textbf{(2.12)}$$

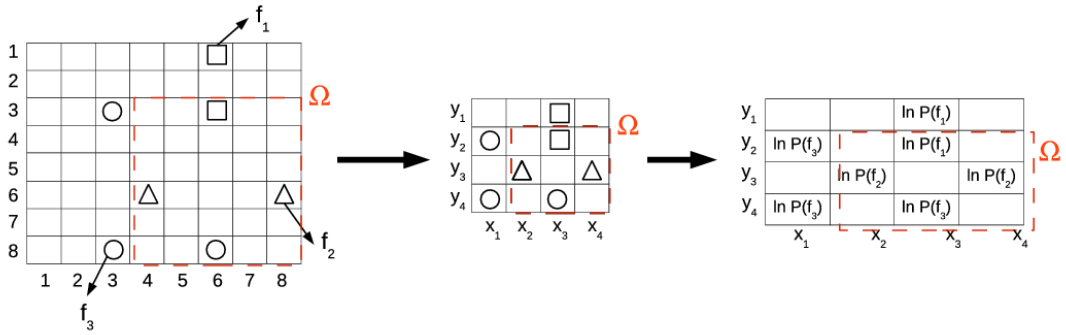

Figure 2.5: The image representation that is adopted to perform the Branch& Bound search efficiently.

The integral images increase the efficiency of the landmark extraction algorithm tremendously. The joint probability term in (**2.3**) normally requires $K(\Omega)$ computations for a given region Ω. Thanks to integral images, this term can constantly be computed with four additions. The joint probability $P(F_\Omega)$ of the features falling into a rectangular region Ω_t which is defined by its top-left and bottom-right coordinates (x_l, y_t) and (x_r, y_b) can simply by calculated as:

$$\ln P(F_\Omega) = II_F(x_r, y_b) - II_F(x_r - 1, y_t - 1) - II_F(x_l - 1, y_b - 1) + II_F(x_l - 1, y_t - 1). \quad \textbf{(2.13)}$$

Thanks to this property of integral images, the output of the energy function that is formulated under the naive-Bayes assumption, (**2.3**), is computed very efficiently. Note that in (**2.3**) and (**2.11**) the logarithms of the probabilities are used rather than the probability values themselves. The reason for employing logarithms is to convert the multiplications in (**2.3**) into additions and this way enable the efficient evaluation of the joint probability term through (**2.13**). An exemplar I_F and II_F is shown in Figure 2.6.

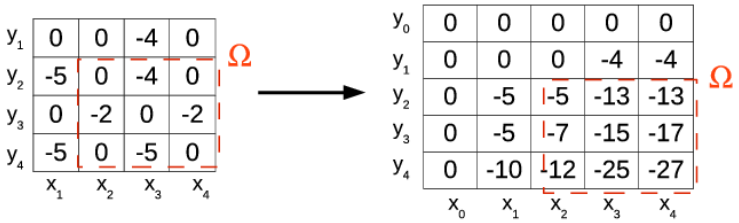

Figure 2.6: An exemplar I_F and II_F.

3. LEARNING AND RE-IDENTIFYING THE LANDMARKS

In this chapter, two important components of the proposed loop closure detection framework have been explained: 1) Learning the appearance of the extracted landmarks; 2) robustly detecting (re-identifying) them. Both components are of major importance since the places are represented by means of those landmarks.

Loop closing is a real-time, on-line process where the appearance of the scene (and the aforementioned set of landmarks) is continuously altered due to perspective changes caused by the camera motion. The machine learning technique, which will be used to learn the landmarks in this context, must possess two attributes. Firstly, the training and testing of the technique must be very fast. Secondly, it must enable updating the object (landmark) model whenever new positive/negative data are present. However, most of the object detection methods proposed by the computer vision community rely on an off-line training phase which requires large training data.

3.1 Learning the Landmarks

Among many successful object detection/recognition techniques, the *ferns* classifier proposed by Ozuysal *et al.* in [37], is the most prominent and adequate one for the needs that have been stressed so far. Its training and test phases are very fast whereby it allows incremental training and it generalizes well with few training instances. Details regarding this method can be found in [37].

In few words, the *ferns* method utilizes very simple features called *ferns*, which consist of several binary tests. These binary tests are nothing more than the comparison of the intensity values of two randomly located pixels — the number of the binary tests and the location of the pixels that are being compared is fixed. The key point of the *ferns* classifier is that it captures the dependencies between these simple tests in a semi-naive Bayesian structure. The pixel comparisons are grouped and each group is referred to as a *fern*. Each *fern* consists of N comparisons, and the total number of *ferns* used for

each classifier is M. The comparisons within each frame are statistically dependent and therefore are evaluated together, however, the ferns are statistically independent among themselves. As it is seen this is a semi-naive Bayesian framework, since the statistical dependencies are considered only inside each *fern*. The mathematical definition and the detailed description of the *ferns* technique is given in [37].

On of the major drawbacks of the *ferns* technique is that it is quite memory consuming, since each *fern* classifier requires an array of $N \cdot 2^M$ real numbers, where N is the number of ferns and M is the number of binary tests per *fern*. In [37] Ozuysal *et al.* discuss that a trade-off can be made between the performance of the classifier and the memory required to store it. Memory efficiency is a key issue in a SLAM application since it can easily be a threat for the scalability of the SLAM application.

In spite of extensive experiments, the parameters of the *fern* classifiers used to describe the landmarks in this study have been picked as $N = 10$ and $M = 13$. As a result each landmark requires 655KB of memory.

The memory consumption problem of *fern* classifiers and their negative effect on the scalability of a SLAM algorithm have also been discussed in [9]. Section 4.2, describes how the memory demands of the *fern* classifier are being dealt with.

The positive and negative data, which are used to train the detector, are chosen around a landmark as it has been depicted in Figure 3.1. To label the samples as positive or negative, the overlap area between the samples and the actual landmark is used. Samples are labeled as positive if the overlap area is large, or they are labeled as negative if the overlap area is small.

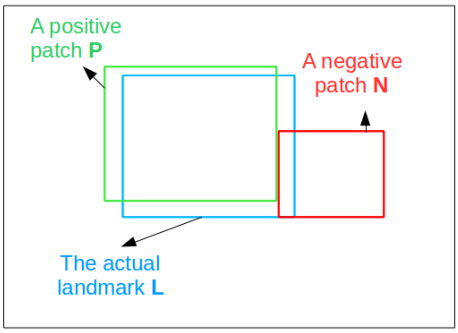

Figure 3.1: An illustration for the selection of the positive and negative samples.

Let Ω_L be the rectangle bounding the landmark, the overlap area ratio between Ω_L and any Ω_i rectangles is defined as $\hat{H}(\Omega_i, \Omega_L) = \min(\frac{H(\Omega_i \cap \Omega_L)}{H(\Omega_L)}, \frac{H(\Omega_i \cap \Omega_L)}{H(\Omega_i)})$. Any rectangle Ω_i is evaluated as a positive sample if $\hat{H}(\Omega_i, \Omega_L) \geq 0.8$ or it is evaluated as a negative sample if $\hat{H}(\Omega_i, \Omega_L) \leq 0.2$. In order to increase the robustness of the detector, positive patches are synthetically warped to obtain additional positive patches as it is suggested in [37, 38].

In order to detect multiple landmarks, a separate classifier is trained for each landmark rather than training a single multi-class classifier. The reasons for training dedicated classifiers are discussed in Section 3.2.

3.2 Detecting the Landmarks

Once the landmarks are described through the *ferns* classifier, this description is used to re-identify the landmarks in other images. However, a search technique is needed to determine the candidate subwindows that are going to be tested.

The search technique that has been adopted to detected the landmarks is the well-known sliding window method [33]. In Section 2.3, it has been discussed that this approach is not efficient to search the salient patches. However, the approach is suitable to detect the landmarks for two reasons. Firstly, in the case of detecting the landmarks, the shape of the rectangle which will be searched is known. Secondly, the landmarks are sought within a pre-defined region of interest (ROI) rather than the whole image.

A dedicated search grid for each landmark is defined. Using a single grid is not possible since the shape of the landmarks is not common. Defining a separate grid for each landmark does not cause an extreme computational overhead since the search space of each landmark is quite limited and the test of the classifier is very fast. The average speed performance of the detector has been given in Section 6.3.

One issue that must be stressed here is that a detection confidence is assigned to the detection hypothesis cast for each landmark. This confidence value is assigned in a very simple fashion, yet it is quite effective.

The sliding window method usually outputs multiple detections around each landmark. It has been observed that true positives tend to be surrounded with more detections comparing to false negatives such as in Figure 3.2. As it is seen, the true positive on the left image is indicated with many detections, whereas a false alarm on the right image is indicated with much fewer detections.

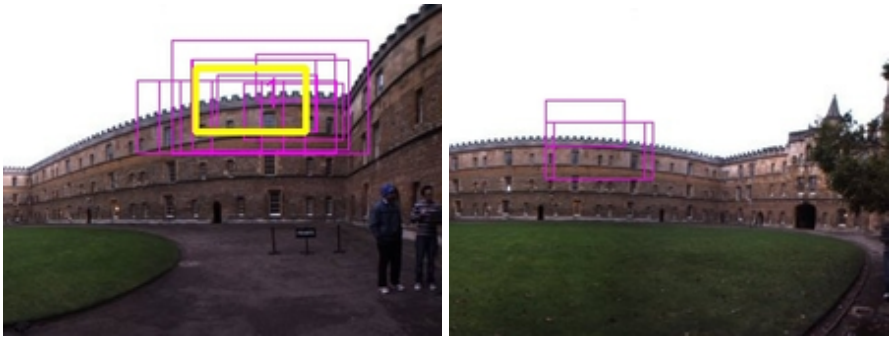

Figure 3.2: Examples to identified landmarks.

The detections that point to more or less the same area are combined by using the overlap area between the detection rectangles. A detection group is formed by the combined rectangles. Let d_t^i indicate the number of detections pointing to landmark l_i at time t. The detection confidence is simply defined as:

$$z_i^t = \frac{d_t^i}{\max(d_1^i, \ldots, d_t^i)}. \tag{3.1}$$

Furthermore, the average detection confidence of each landmark l_i is also computed at each time-step t as $\bar{z}_i^t$:

$$\bar{z}_i^t = \frac{\sum_{j=t_i}^{t} d_i^j}{t - t_i + 1}, \tag{3.2}$$

where t_i is the time that the i^{th} landmark is extracted.

At this point, it must be stressed that false positives are tolerated at the training phase up to a reasonable extent. The landmark description becomes more generalized in this case and the number of false alarms increases substantially. On the other hand, the detections around the actual landmark turn out even more evident. The landmark detection hypothesis adopted in this study is not a binary decision — as explained earlier, each detection hypothesis is cast along with a confidence value. Landmark detection hypotheses cast at different confidence levels are meant to stand for different image patches. In other words, the same landmark model is used to describe multiple

patches, and the distinction among these patches is made by the detection confidence of each hypothesis. This way, the landmarks are "shared" among several patterns. This lowers the memory requirements of the loop closure detection system dramatically. The memory consumption of the fern classifiers had been discussed back in Section 3.1; obviously less landmarks are better for the memory. The number of landmarks extracted during the test on the 2.2km long New College Dataset (8127 images) [28] is as low as 266. The size of the landmark database has been discussed and also its variation over the time has been illustrated in Section 4.1.

4. CONSTRUCTING THE APPEARANCE SPACE VIA LANDMARKS

The previous chapters have described the methodology adopted to extract visual landmarks, the methodology used to learn the appearance of them and re-identify them in subsequent scenes. The ultimate purpose of the landmark extraction and learning procedures is constructing a sparse appearance space, where locations will sparsely be modeled and loop closure estimations will be cast. This chapter explains how to use the landmarks in order to define this appearance space.

The rest of the chapter is organized as follows: Section 4.1 introduces the landmark database and how explains how it is constructed and updated, and Section 4.2 introduces the sparse location model which is built using the landmarks.

4.1 The Landmark Database

In Section 1.3 it has been declared that the landmark database which is used to recognize places, consists of landmarks that are specific to the environment that the robot navigates in. Therefore, the database in question is initially empty and updated on-line throughout the trajectory. New landmarks are appended to the database as new locations are being traversed — recall that, locations are represented with multiple landmarks as explained and illustrated in Section 2.2. Let $\mathcal{L}^t$ be the landmark database consisting of the landmarks extracted up to time t. As stressed earlier, the landmark database is updated on-ine throughout the trajectory such as $\emptyset \subseteq \mathcal{L}^0 \ldots \subseteq \mathcal{L}^{t-1} \subseteq \mathcal{L}^t$.

However, the landmarks in the existing database are searched before new landmarks are appended; if sufficient landmarks are detected on the incoming image, new landmarks are not created. This upper limit on the landmarks is denoted with B, and it is enforced for two reasons. Firstly, as discussed in Section 3.1, each landmark occupies a considerable amount of memory; therefore, the size of the landmark database must be restraint if long-term operation is desired. Secondly, experimental results indicate that the performance of the overall system doesn't increase after a

certain number of landmarks. The pseudo-code of the landmark database update procedure has been given in the following algorithm.

Algorithm 2 Updating $\mathcal{L}$ at time t

Require: Landmark database $\mathcal{L}$
Require: Image at time t, I_t
  Search for existing landmarks $\mathcal{L}^{t-1}$ in I_t: $\mathbf{l_t}$
  for $i = 0$ to $max(0, B - |\mathbf{l_t}|)$ **do**
    Detect new salient region $\Omega^* = \text{argmax}_{\Omega \in I_t} E(F_\Omega, \Omega)$
    Learn new landmark l out of region Ω^*
    Append l to landmark database $\mathcal{L}^t$
  end for

It must be noted that inconsistent landmarks, *i.e.* landmarks which are not detected in the next 3 frames, are removed from the database immediately.

As it has been stressed earlier, the variation of $\mathcal{L}$ with respect to t is an important factor for the overall system, since it has a direct influence on the scalability of the loop closure detector. This variation has been illustrated for the two test runs that have been performed on two datasets: 1) The New College Dataset [28], which is an outdoor dataset consisting of 8127 images, 2) the İTÜ Robotics Laboratory dataset, which is an indoor dataset consisting of 2400 images. The change in $\mathcal{L}$ with respect to time, has been illustrated in Figure 4.1 for both of these datasets.

The top plot in Figure 4.1 demonstrates the variation of $|\mathcal{L}|$ w.r.t. time on the New College Dataset. At the end of the test run, the size of $|\mathcal{L}|$ is as small as 266, which occupies 173 MB of memory in RAM — each landmark occupies 655 KB, see Section 3.1.

The change of $|\mathcal{L}|$ w.r.t. time on the experiments performed on the indoor dataset of İTÜ Robotics Laboratory is present on the lower plot in Figure 4.1. The final size of $\mathcal{L}$ on this test is 168. The test on this dataset involves a circular trajectory which is traversed twice. The frames $\{0, 1, \ldots, 1200\}$ are collected at the first run and the frames $\{1201, 1202, \ldots, 2400\}$ are collected at the second run. This implies that the first run does not contain any loop closure; and in contrary to the first run, each frame collected in the second run has a match from the first run. In other words, the second run is the test run and each frame in this test has a match from the first set a loop closure hypothesis must be cast. The fact that the second half is the test run is quite

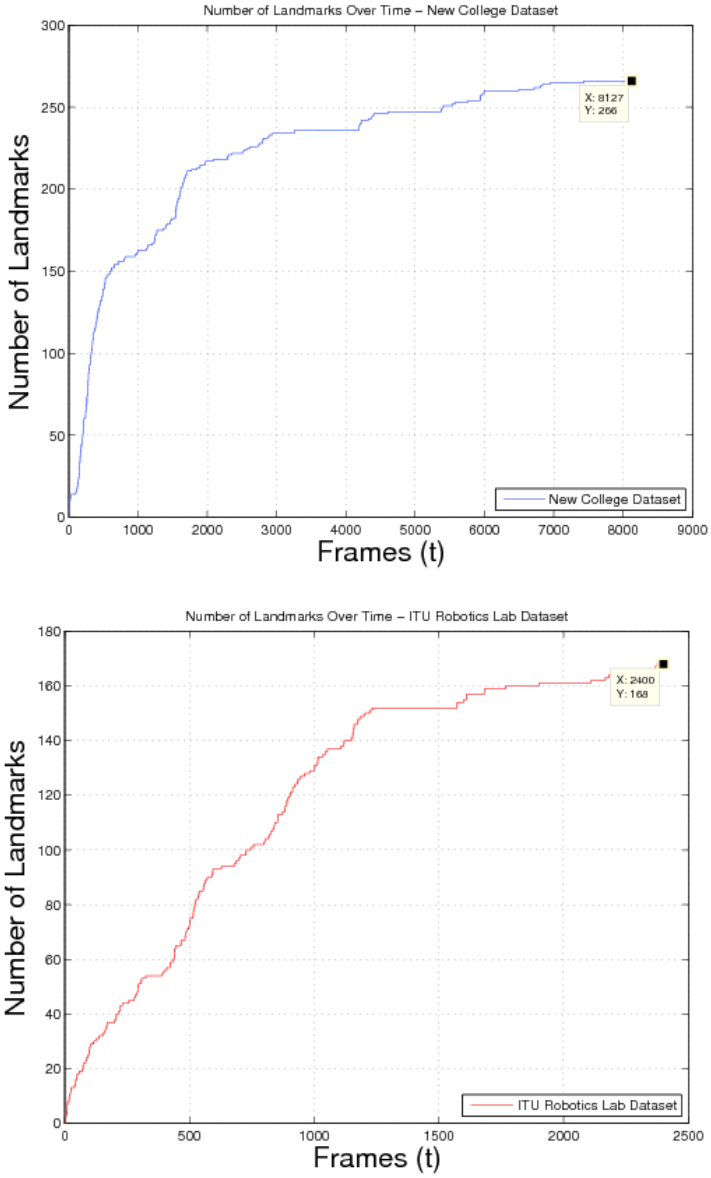

Figure 4.1: The change in the size of landmark database with respect to time.

evident in the plot: The slope of the curve part corresponding to the frames $0 - 1200$ is much higher than the slope of the curve part corresponding to the frames $1200 - 2400$. In other words, the landmark database is updated much more frequently when new locations are being visited, whereas it is updated much more rarely when previously visited locations are being re-visited.

4.2 The Location Model

The main purpose of this thesis is modeling place images sparsely through landmarks in order to assess the similarity between images and detect loop closures. The previous

chapters have described how the landmarks are extracted and learnt. However, it has not been explained how these landmarks are used as part of the location model. In other words, it has not been discussed which properties of the detected landmarks are used (*e.g.* their spatial location, detection confidence etc.), when the location model is being created .

In spite of the information described in Section 3.2, each landmark l_i^t detected at time t, comes along with four properties:

- The identity of the landmark i.
- The spatial coordinates of the landmark $\mathbf{x}_i^t = \begin{bmatrix} x_i^t & y_i^t \end{bmatrix}$. These coordinates stand for the 2D point of the center of the landmark on the image plane.
- The detection confidence of the landmark z_i^t, which is computed out of (**3.1**).
- The size of the landmark.

Three of these four properties are being considered in the location model: The identity of the landmark, the spatial location of the landmark and the detection confidence of the landmark. The scale is simply being discarded.

Each landmark l_i at time t is described as $l_i^t = \{z_i^t, \mathbf{x_i^t}\}$. The coordinates of each landmark $\mathbf{x}_i^t$ are defined for all t, even if l_i is not detected at this time — they are simply set to zero in the case that the landmark is not detected. The location model has been illustrated in Figure 4.2.

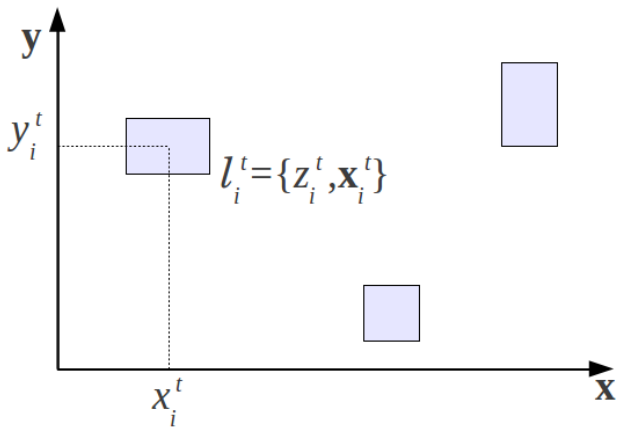

Figure 4.2: An illustration of location representation.

Figure 4.2 depicts the toy representation of an exemplar location model with 3 landmarks. The x and y axes define the image plane. The landmarks have been shown with the shaded rectangles. The information used to describe each landmark in the context of the location model has been shown on an exemplar landmark l_i^t.

4.3 Constructing the Appearance Space

The appearance space, which contains the images of all visited locations, is defined as $\mathbf{I}^t = \{I_0, I_1, \ldots, I_t\}$. Any image acquired at time t is sparsely represented with $I_t \equiv \{Z_t, X_t\}$ where Z_t contains the detection confidences and X_t the spatial coordinates of all the landmarks up to time t:

$$\mathbf{Z}_t = \begin{bmatrix} z_1^t & \ldots & z_{k_t}^t \end{bmatrix} \tag{4.1}$$

$$\mathbf{X}_t = \begin{bmatrix} \mathbf{x}_1^t & \ldots & \mathbf{x}_{k_t}^t \end{bmatrix} \tag{4.2}$$

where $k_t = |\mathcal{L}^t|$.

The appearance space $\mathbf{I^t}$ is continuously updated, as new images are being acquired from the camera.

5. LOOP CLOSURE DETECTION ON THE APPEARANCE SPACE

The majority of the components that together form the loop closure system described in this thesis have been explained in the previous chapters. The extraction, learning and identification stages of the landmarks have been explained. Furthermore, the construction of location models and eventually the appearance space has also been described.

This chapter focuses on explaining how the loop closures are ultimately detected on the appearance space. Loop closure detection is basically achieved by measuring the similarity between the image of the current location I_t and all previously visited locations, that lie on the appearance space $\mathbf{I^t}$ (see Section 4.3).

There are two major issues that must be tackled in order to detect loop closures: 1) The similarity between two images must be measured 2) previously visited locations must somehow be revealed, in order to understand whether a place has been seen before or not. This chapter addresses these issues; Section 5.1 describes the similarity criterion used to assess the similarity between two images and Section 5.2 explains how the unseen locations are revealed.

5.1 Measuring the Similarity Between Locations

In order to find out the similarity between two locations, a straightforward function Ψ has been used to measure the similarity between their images I_{t_1}, I_{t_2} such as:

$$\Psi(I_{t_1}, I_{t_2}) = \sum_{i=1}^{k_{min}} \bar{z}_i \left| z_i^{t_1} - z_i^{t_2} \right| \left\| \mathbf{x}_i^{t_1} - \mathbf{x}_i^{t_2} \right\|, \qquad \textbf{(5.1)}$$

where $\left|\cdot\right|$ stands for the absolute value operator, $\left\|\cdot\right\|$ stands for the l^2-norm and $k_{min} = \min(|\mathcal{L}^{t_1}|, |\mathcal{L}^{t_2}|)$. Setting the upper limit of the sum in (**5.1**) to k_{min} ensures a fair comparison between the observations — only the landmarks which existed at time $t_{min} = \min(t_1, t_2)$ are being taken into account. The reason for that is these are

the landmarks that were commonly sought in both images. It is noteworthy to state that all loop closure detection techniques don't necessarily utilize the spatial location information of the features. The well-established FAB-MAP technique [3] for instance discards the location information and simply utilizes the binary detection output of the features.

The recognition of a previously visited location at time t is achieved by matching the current image I_t with its best match within the appearance spce $\mathbf{I^t}$ by using the similarity criterion in (**5.1**):

$$I_{t^*} = \underset{I_j \in \mathbf{I^{t-1}}}{\operatorname{argmin}} \Psi(I_t, I_j). \tag{**5.2**}$$

The maximization in (**5.2**) may output a correct result only if the location at time t has been seen before. The next section, explains how it is managed to reveal whether a place has been visited throughout the trajectory.

5.2 Determining Unseen Locations

A fact that must be considered in the design of a loop closure detection system is that how it will be revealed whether an observation comes from an unseen location or not. In this section it is being described how it has been tackled with this fact.

Let Y_t be a signal vector of length $t-1$, which contains the similarity scores between the incoming image and all the images that lie inside the appearance space $\mathbf{I^{t-1}}$: $Y_t = \begin{bmatrix} \Psi(I_t, I_0) & \dots & \Psi(I_t, I_{t-1}) \end{bmatrix}$. This signal is produced in order to reveal whether the current location has been visited before or not. The local part of Y_t around the best match exhibits a clear peak when I_t is the image of a previously visited location.

In order to evaluate the local signal around the best match $\tilde{Y}_{t^*}$, the original signal Y_t is altered twofold: It's cropped around the best match I_{t^*} and it's normalized (z-normalization):

$$\tilde{Y}_{t^*} = \begin{bmatrix} \frac{\Psi(I_{t^*}, I_{t^*-\Delta t}) - \mu_{\tilde{Y}_{t^*}}}{\sigma_{\tilde{Y}_{t^*}}} & \dots & \frac{\Psi(I_{t^*}, I_{t^*+\Delta t}) - \mu_{\tilde{Y}_{t^*}}}{\sigma_{\tilde{Y}_{t^*}}} \end{bmatrix}, \tag{**5.3**}$$

where $\mu_{\tilde{Y}_{t^*}}$ and $\sigma_{\tilde{Y}_{t^*}}$ are respectively the mean and standard deviation of $\tilde{Y}_{t^*}$ defined as follows:

$$\mu_{\tilde{Y}_{t^*}} = \sum_{k=t^*-\Delta t}^{t^*+\Delta t} \frac{\Psi(I_{t^*}, I_{t^*+k})}{2\Delta t + 1} \tag{5.4}$$

$$\sigma_{\tilde{Y}_{t^*}} = \sqrt{\sum_{k=t^*-\Delta t}^{t^*+\Delta t} \frac{\Psi(I_{t^*}, I_{t^*+k}) - \mu_{\tilde{Y}_{t^*}}}{2\Delta t + 1}} \tag{5.5}$$

As it has been stressed earlier in this section, the normalized local similarity signal $\tilde{Y}_{t^*}$ exhibits valuable information which can be used to understand whether the best matching location image I_{t^*} has really been seen before or not. To achieve this, the peak of the the normalized signal $\tilde{Y}_{t^*}$ is compared to a predefined threshold θ. If it exceeds this threshold (if $\max(\tilde{Y}_{t^*}) > \theta$), it is assumed that the matched location is seen before and the loop closure hypothesis is finally cast. The Δt value, which is used to crop the signal, must be defined in accordance with the average speed of the vehicle.

Exemplar $\tilde{Y}_{t^*}$ signals have been shown in Figure 5.1. The plots in the upper row of the figure, depict cases where a loop closure exists, and the plots in the lower row of the figure depict cases where a loop closure does not take place. It is eminent that in the case of actual loop closure existence, the signal exhibits a clear, distinctive peak. On the other hand, the signal is quite scattered when a loop closure does not exist. The green lines on the plots stand for the threshold value θ, which is fixed for all cases since the signal is normalized.

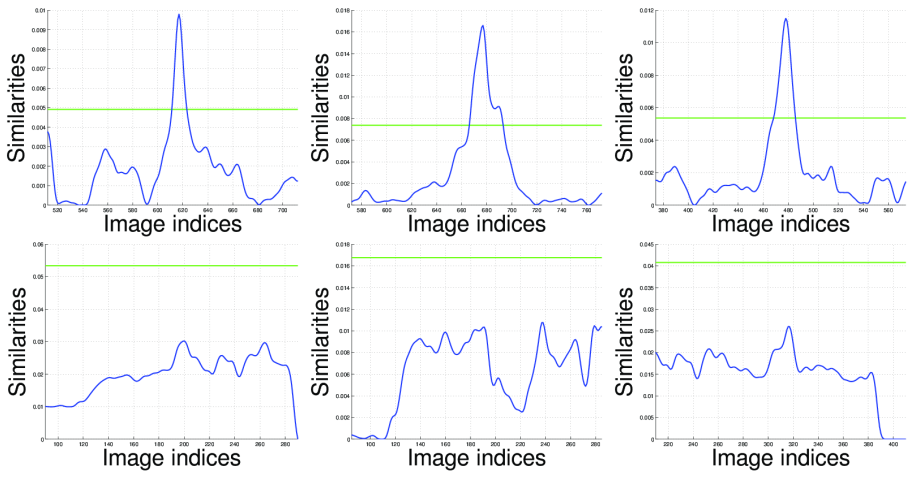

Figure 5.1: Exemplary normalized local similarity signals.

6. EXPERIMENTAL RESULTS

The previous chapters have introduced all of the components of the proposed loop closure detection system. This chapter gives the evaluation of the system, through the experiments conducted on two datasets.

6.1 Experimental Setup

The proposed loop closure system has been tested on the following datasets: 1) The New College dataset presented in [28], 2) A dataset collected inside the Robotics Laboratory of Istanbul Technical University.

The New College dataset is a dataset consisting of 8127 images collected with a panoramic camera mounted on a mobile robot. The ground truth of this data is extracted out of the GPS information. However, the GPS signal is frequently interrupted; therefore, the ground truth is provided for only the 3553 of these frames (approximately the 44% of the dataset). The dataset has been collected on the New College campus of Oxford University and the length of path traversed during the dataset collection is 2.2km.

The second dataset, İTÜ Robotics Laboratory dataset is an indoor dataset consisting of 2400 images collected with a hand-held camera. A circular trajectory has been traversed in order to collect this dataset has been collected. The first tour on this circle, naturally does not contain any loop closure, whereas each frame collected in the second tour has a corresponding match from the first one. In this case, the ground truth data used to evaluate the loop closures estimations is easily formed. The loop closure scenarios, however, in this dataset involve image matches where considerable rotation and translation is present, since the dataset has been collected with a hand-held camera.

The experimental results on these two datasets which exhibit quite different characteristics, indicate that the proposed loop detection closure detection system is subject to perform well under various conditions.

The C++ implementation of the proposed method and the indoor İTÜ Robotics dataset is available on `http://www.robotics.itu.edu.tr/slam`. The video result of the test carried out on the New College dataset is also available on the same link.

6.2 Loop Closure Detection Performance

The metric that is mostly used to evaluate the loop closure detection systems is the precision-recall curve. The outcome of the experiments have been evaluated through this metric. In Section 1.1, it has been highlighted that the false loop closures might turn out catastrophic for the overall SLAM system. Therefore, the recall rates at high precision are much more significant for the system.

The precision-recall curves obtained on the tests performed on both datasets are shown in Figure 6.1. The curves were obtained by manually adjusting the threshold at which the loop closure is determined.

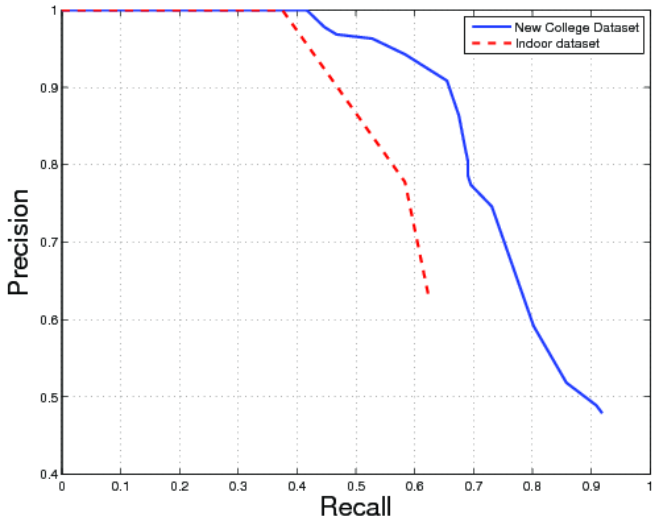

Figure 6.1: The precision-recall curves of the method on two datasets.

Some examples of loop closure detections on the New College datasethave been shown in Figure 6.2. The only false alarm on a test run on the New College dataset has been shown in the last image of the bottom row of Figure 6.2.

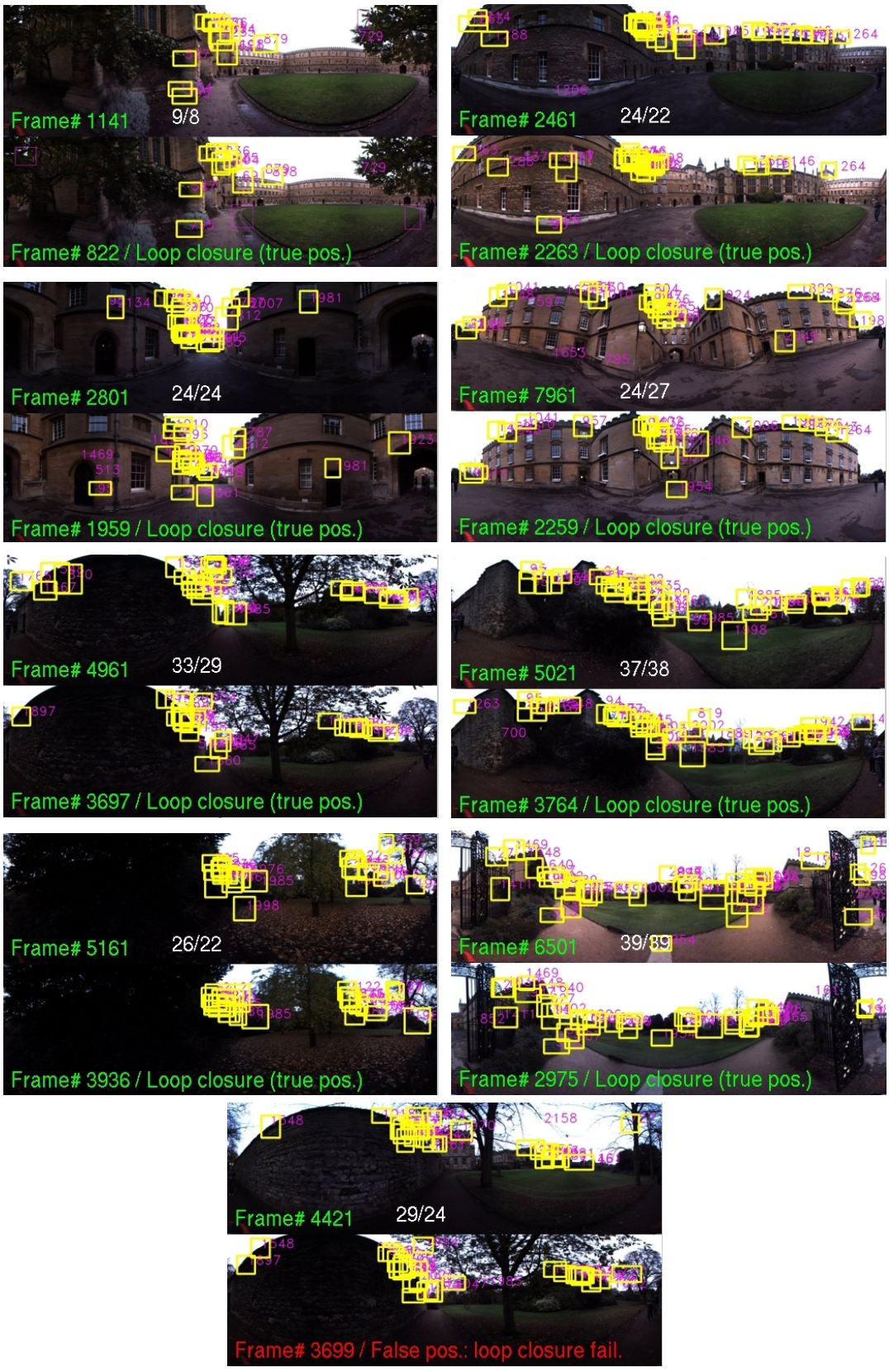

Figure 6.2: Some examples of matched image pairs from the New College dataset.

The proposed loop closure method is able to detect loop closures with 26.4% recall rate on the New College dataset at 100% precision, and 31.5% recall rate at 99% precision. According to [39] and [40], the FAB-MAP technique attains 12% recall rate at 100% precision and 16% recall rate at %99.6 precision on this dataset. The results on the outdoor dataset indicate that the scalability of the method is promising.

The proposed technique attains 37% precision at 100% precision on the indoor dataset of İTÜ Robotics Laboratory, which contains images collected with a handheld camera, there is an observable view difference caused by rotation and translation. This dataset contains exibits clear translationand rotation since it is collected with a handheld camera. It also exhibits strong perceptual aliasing. The precision-recall curve on this dataset is shown in Figure 6.1. Exemplary loop closure results have been shown in Figure 6.3.

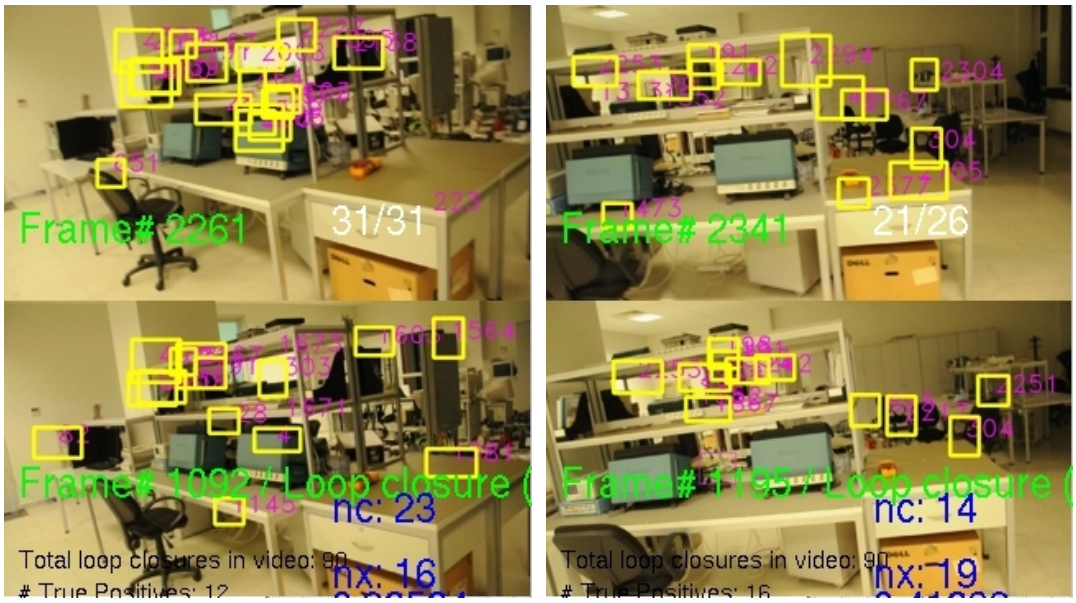

Figure 6.3: Some examples of matched image pairs from the İTÜ Robotics Laboratory dataset.

6.3 Speed Performance of the Method

Computational efficiency is a key issue for a loop closure detection technique since loop closing is an on-line process. The proposed technique includes several components: Multiple landmark extraction, learning, and detection followed by image matching. The average processing time of each component has been reported for

images of size 1024×309 in Table 6.1 — the tests were performed on a Intel Core 2 Duo 2.2GHz CPU.

Table 6.1: Speed performance of the method

Task	Average Time
Extraction and clustering of SURF features	780ms
Extraction of a salient patch (landmark)	2.5ms
Learning a landmark	5ms
Learning a landmark (with warped samples)	850ms
Detection of a landmark	1.75ms

The processing time of landmark detection in Table 6.1 has been given for a single landmark. The total number of landmarks by the end of the test performed on the New College Dataset was 354, and the average time spent to search all of these landmarks is 645ms per frame. The overall average processing time for an image is 2.64 seconds. Most of this time is spent to the extraction of SURF features of the BoW model. However, the SURF features may be replaced with other features since they are only used for saliency detection and not location representation.

The second most time consuming process involves the training of the landmarks. Most of the computational effort is spent during the warping process of the positive and negative samples. The computational overhead of the overall training process turns out negligible when patches are not warped. In a future work, we consider to warp the *fern* features instead of images, in order to avoid this computational overhead.

7. CONCLUSIONS AND FUTURE WORK

In this thesis, a new vision based loop closure detection system has been developed. The problem of loop closure detection has been considered as a part of the SLAM problem, and the proposed system has been developed within this context. The developed system, however, does not depend on the SLAM estimations.

7.1 Conclusions

As it has been explained in the introduction chapter, loop closure problem is a very complex problem, which is challenging in many aspects. All of the following criteria are major issues of concern in the context of the loop closing problem: Accuracy, efficiency and scalability. Developing an accurate system is difficult for two reasons. Firstly, certain variations that alter the image appearance dramatically, *e.g.* camera frame translation/rotation, illumination and view perspective. Secondly, the perceptual aliasing problem, which is very typical for many kinds of environments (forests, offices, urban regions etc.), makes the accurate detection of correct loop closures even more challenging.

The main idea of the proposed system is representing the locations sparsely through visual landmarks. These locations define an appearance space, and the loop closure estimations are ultimately cast on this appearance space.

The loop closure detection scheme that has been described, involves several problems that must be dealt with: 1) Unsupervised landmark extraction, 2) learning the appearance of the landmarks in order to re-identify them, 3) constructing an appearance space through the landmarks, and assessing the similarity among the images of this space. The solutions adopted to solve each of these problems has been described in the previous chapters.

The motivation behind the usage of landmarks for loop closure detection, is that many living beings, including humans, successfully use visual landmarks to describe locations and navigate in a topological manner. Several techniques that utilize local image representation have been proposed before; however, most of these techniques consider a database which is built off-line and consists of small generic features fixed in size. In contrast to these techniques, the system presented in this thesis builds a database on-line. This database consists of visual landmarks which vary in size. More importantly, the landmarks are specific to the environment that the robot is traveling.

The main contributions of this thesis are twofold. The first contribution is a saliency detection technique, which in the context of this study has been used to extract the visual landmarks. The second contribution, is an overall loop closure detection scheme where images are matched on an appearance space using a similarity metric.

It has been demonstrated that the performance of the presented system is comparable to the state of the art to say the least. On the other hand, its speed performance is promising, even though real-time operation is not possible at this point. The system may however perform in a soft-real time scheme. The bottleneck of the system, is the SURF feature computation step which takes place during the BoW model preparation. Fortunately, the computation of SURF features is not a key component of the proposed system. The BoW model constructed from SURF features can be replaced with any kind of representation that allows statistical inference out of its features. The most straightforward solution towards a more efficient system, is to use the BoW model by replacing the SURF features with simpler and faster features.

7.2 Future Work

The system presented in this thesis describes a novel loop closing system, which is subject to be implemented on a real-life SLAM application. However, before it can be implemented on an actual SLAM system, it requires several improvements.

One of the most crucial improvements that should be carried out is increasing the speed of the system overall. This can be achieved either by parallelizing the algorithm (on a GPU or a more powerful CPU), or, as stated earlier, by increasing the efficiency

of the method via replacing the SURF features with more efficient ones. Increasing the efficiency of the technique, is naturally a more feasible and desirable solution. However, it must be shown, through experiments, that the features that replace the SURF features are at least as efficient as they are.

The second most significant improvement that can be made, is replacing the approach that determines unseen locations with a more robust one. The current approach utilizes the local similarity values around the best match. However, this method requires the speed information of the vehicle; therefore, the system depends on the velocity estimations of the mobile robot. Moreover, solving the *kidnapped robot* problem is not possible with the current approach.

REFERENCES

[1] **Ho, K. and Newman, P.** (2007). Detecting Loop Closure with Scene Sequences, *International Journal of Computer Vision*, ***74***, 261–286, 10.1007/s11263-006-0020-1.

[2] **Newman, P.**, **Cole, D. and Ho, K.** (2006). Outdoor SLAM using visual appearance and laser ranging, *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp.1180 –1187.

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

[4] **Newman, P. and Ho, K.** (2005). SLAM-Loop Closing with Visually Salient Features, *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pp.635 – 642.

[5] **Bosse, M.**, **Newman, P.**, **Leonard, J.**, **Soika, M.**, **Feiten, W. and Teller, S.** (2003). An Atlas framework for scalable mapping, *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 2, pp.1899 – 1906 vol.2.

[6] **Angeli, A.**, **Doncieux, S.**, **Meyer, J.A. and Filliat, D.** (2008). Incremental vision-based topological SLAM, *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp.1031 –1036.

[7] **Williams, B.**, **Cummins, M.**, **Neira, J.**, **Newman, P.**, **Reid, I. and Tardós, J.** (2009). A comparison of loop closing techniques in monocular SLAM, *Robotics and Autonomous Systems*.

[8] **Clemente, L.**, **Davison, A.**, **Reid, I.**, **Neira, J. and Tardos, J.** (2007). Mapping Large Loops with a Single Hand-Held Camera, *Robotics: Science and Systems*.

[9] **Williams, B.**, **Cummins, M.**, **Neira, J.**, **Newman, P.**, **Reid, I. and Tardos, J.** (2008). An image-to-map loop closing method for monocular SLAM, *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pp.2053 –2059.

[10] **Ulrich, I. and Nourbakhsh, I.** (2000). Appearance-based place recognition for topological localization, *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 2, pp.1023 –1029 vol.2.

[11] **Lamon, P.**, **Nourbakhsh, I.**, **Jensen, B. and Siegwart, R.** (2001). Deriving and matching image fingerprint sequences for mobile robot localization, *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, volume 2, pp.1609 – 1614 vol.2.

[12] **Torralba, A.**, **Murphy, K.**, **Freeman, W. and Rubin, M.** (2003). Context-based vision system for place and object recognition, *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pp.273 –280 vol.1.

[13] **Kröse, B.**, **Vlassis, N.**, **Bunschoten, R. and Motomura, Y.** (2001). A probabilistic model for appearance-based robot localization, *Image and Vision Computing*, ***19***(6), 381 – 391.

[14] **Ramos, F.**, **Upcroft, B.**, **Kumar, S. and Durrant-Whyte, H.** (2012). A Bayesian approach for place recognition, *Robotics and Autonomous Systems*, ***60***(4), 487 – 497.

[15] **Bowling, M.**, **Wilkinson, D.**, **Ghodsi, A. and Milstein, A.**, (2007). Subjective Localization with Action Respecting Embedding, Robotics Research, volume 28 of *Springer Tracts in Advanced Robotics*, Springer Berlin / Heidelberg, pp.190–202.

[16] **Lowe, D.G.** (2004). Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision*, ***60***, 91–110, 10.1023/B:VISI.0000029664.99615.94.

[17] **Bay, H.**, **Tuytelaars, T. and Van Gool, L.**, (2006). SURF: Speeded Up Robust Features, Computer Vision – ECCV 2006, volume3951 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp.404–417.

[18] **Sivic, J. and Zisserman, A.** (2003). Video Google: a text retrieval approach to object matching in videos, *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pp.1470 –1477 vol.2.

[19] **Li, F. and Kosecka, J.** (2006). Probabilistic location recognition using reduced feature set, *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pp.3405 –3410.

[20] **Ferreira, F.**, **Santos, V. and Dias, J.** (2006). Integration of Multiple Sensors using Binary Features in a Bernoulli Mixture Model, *Multisensor Fusion and Integration for Intelligent Systems, 2006 IEEE International Conference on*, pp.104 –109.

[21] **Cadena, C.**, **Galvez-Lopez, D.**, **Ramos, F.**, **Tardos, J. and Neira, J.** (2010). Robust place recognition with stereo cameras, *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pp.5182 –5189.

[22] **Angeli, A.**, **Filliat, D.**, **Doncieux, S. and Meyer, J.A.** (2008). Fast and Incremental Method for Loop-Closure Detection Using Bags of Visual Words, *Robotics, IEEE Transactions on*, ***24***(5), 1027 –1037.

[23] **Chow, C. and Liu, C.** (1968). Approximating discrete probability distributions with dependence trees, *Information Theory, IEEE Transactions on*, ***14***(3), 462 – 467.

[24] **Fooa, P.**, **Warrena, W.H.**, **Duchona, A. and Tarra, M.J.** Do Humans Integrate Routes Into a Cognitive Map? Map- Versus Landmark-Based Navigation of Novel Shortcuts, *Journal of Experimental Psychology: Learning, Memory, and Cognition*, ***31***(2), 195–215.

[25] **Collett, T.S.** (1996). Insect navigation en route to the goal: Multiple strategies for the use of landmarks, *Journal of Experimental Biology*, ***199***, 227–235.

[26] **Biegler, R.** (2000). Possible uses of path integration in animal navigation, *Animal Learning Behavior*, ***28***(3), 257–277.

[27] **Lampert, C.H.**, **Blaschko, M.B. and Hofmann, T.** (2009). Efficient Subwindow Search: A Branch and Bound Framework for Object Localization, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ***31***, 2129–2142.

[28] **Smith, M.**, **Baldwin, I.**, **Churchill, W.**, **Paul, R. and Newman, P.** (2009). The New College Vision and Laser Data Set, *Int. J. Rob. Res.*, ***28***, 595–599.

[29] **Sariyanidi, E. and Temeltas, H.** (2012). Unsupervised Visual Landmark Extraction For Place Recognition, *SPIE Defense, Security and Sensing*, Baltimore, USA.

[30] **Sariyanidi, E.**, **Sencan, O. and Temeltas, H.** (2012). An Image-to-image Loop-Closure Detection Method Based on Unsupervised Landmark Extraction, *IEEE Intelligent Vehicle Symposium*, Alcala de Henares, Spain.

[31] **Nister, D. and Stewenius, H.** (2006). Scalable Recognition with a Vocabulary Tree, *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pp.2161 – 2168.

[32] **Sparck Jones, K.**, (1988). Document retrieval systems, chapterA statistical interpretation of term specificity and its application in retrieval, Taylor Graham Publishing, London, UK, UK, pp.132–142.

[33] **Rowley, H.**, **Baluja, S. and Kanade, T.** (1998). Neural network-based face detection, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, ***20***(1), 23 –38.

[34] **Land, A.H. and Doig, A.G.** (1960). An Automatic Method of Solving Discrete Programming Problems, *Econometrica*, ***28***(3), 497–520.

[35] **Viola, P. and Jones, M.** (2001). Rapid object detection using a boosted cascade of simple features, *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pp.I–511 – I–518 vol.1.

[36] **Crow, F.C.** (1984). Summed-area tables for texture mapping, *SIGGRAPH Comput. Graph.*, ***18***, 207–212.

[37] **Ozuysal, M.**, **Fua, P. and Lepetit, V.** (2007). Fast Keypoint Recognition in Ten Lines of Code, *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, ***0***, 1–8.

[38] **Lepetit, V. and Fua, P.** (2006). Keypoint Recognition using Randomized Trees, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, ***28***(9), 1465–1479.

[39] **Maddern, W.**, **Milford, M. and Wyeth, G.** (2011). Continuous appearance-based trajectory SLAM, *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, IEEE, pp.3595–3600.

[40] **Newman, P.**, **Chandran-Ramesh, M.**, **Cole, D.**, **Cummins, M.**, **Harrison, A.**, **Posner, I. and Schroeter, D.** (2007). Describing, Navigating and Recognising Urban Spaces - Building An End-to-End SLAM System, *Proc. of the Int. Symposium of Robotics Research (ISRR)*, Hiroshima,Japan.

CURRICULUM VITAE

Name Surname: Evangelos Sarıyanidi

Place and Date of Birth: İstanbul, 1986

Address: Kontrol Mühendisliği Bölümü, İTÜ Ayazağa Kampüsü, 34469 İstanbul

E-Mail: sariyanidi@itu.edu.tr

B.Sc.: Istanbul Technical University, Faculty of Electrical and Electronic Engineering, Control Engineering, 2009

Professional Experience and Rewards:
National Scholarship for M.Sc. Students, The Scientific and Technological Research Council of Turkey (TÜBITAK)
IEEE Best Student Paper Award (1^{st} place) on SIU 2012
Alper Atalay Award (3^{rd} place) on SIU 2012

List of Publications and Patents:

▪ **Sarıyanidi E.**., Tek S. C., and Gökmen M. (2011). Efficient Face Detection using Coarse Sampling. *IEEE 19th Conference on Signal Processing and Communications Applications (SIU 2011)*, 20-22 April, Antalya, Turkey.

▪ **Sarıyanidi E.**, Dağlı V., Tek S. C., Tunç B., and Gökmen M. (2012). A Novel Face Representation using Local Zernik Moments. *IEEE 20th Conference on Signal Processing and Communications Applications (SIU 2012)*, 18-20 April, Muğla, Turkey.

▪ **Sarıyanidi E.**, Dağlı V., Tek S. C., Tunç B., and Gökmen M. (2012). Local Zernike Moments: A New Representation for Face Recognition. *IEEE 19th International Conference on Image Processing (ICIP)*, To appear.

PUBLICATIONS/PRESENTATIONS ON THE THESIS

▪ **Sarıyanidi E.**, Sencan, O. and Temeltas, H. (2012). An Image-to-image Loop-Closure Detection Method Based on Unsupervised Landmark Extraction. *12th Intelligent Vehicle Symposium, 2012*, 3-7 June, Alcala de Henares, Spain.

▪ **Sarıyanidi E.**, and Temeltas, H. (2012). Unsupervised Visual Landmark Extraction For Place Recognition. *SPIE Defense, Security and Sensing* Baltimore, USA.