Place recognition based visual localization in changing environments

Yongliang Qiao

SPIM

Thèse de Doctorat

école doctorale **sciences pour l'ingénieur et microtechniques**

UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

Place recognition based visual localization in changing environments

Yongliang Qiao

SPIM

Thèse de Doctorat

école doctorale **sciences pour l'ingénieur et microtechniques**

UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

N° | X | X | X |

THÈSE présentée par

YONGLIANG QIAO

pour obtenir le

Grade de Docteur de
l'Université de Technologie de Belfort-Montbéliard

Spécialité : **Informatique**

Place recognition based visual localization in changing environments

Unité de Recherche :
Laboratoire Électronique, Informatique et Image

Soutenue publiquement le 3 Avril 2017 devant le Jury composé de :

LOUAHDI KHOUDOUR	Rapporteur	Directeur de Recherche au CEREMA
ABDELMALIK TALEB-AHMED	Rapporteur	Professeur des Universités à l'UVHC
YASSINE RUICHEK	Directeur de thèse	Professeur des Universités à l'UTBM
CINDY CAPPELLE	Co-encadrante	Maître de Conférences à l'UTBM
CYRIL MEURIE	Examinateur	Chargé de Recherche à l'IFSTTAR
PIERRE GOUTON	Examinateur	Professeur des Universités à l'Université de Bourgogne

Acknowledgements

I would like to thank my supervisors Prof. Yassine RUICHEK and associate Prof. Cindy CAPPELLE, for the continuous support and guidance on improving my research. Working with them, I learned a lot both scientific paper writing and critical thinking, both of which are important to my future projects. I am grateful to them for the help they have given me. With their research experience, they teach me how to become an independent researcher.

I would like to give my gratitude to the financial support from the program of China Scholarship Council (CSC). The CSC program is really helpful which gave me a chance to study abroad.

I would like to express my gratitude to Professor Louahdi KHOUDOUR and Professor Abdelmalik TALEB-AHMED, for reviewing this thesis and posing insightful questions. I would like to extend my appreciation to Mr. Cyril MEURIE and Mr. Pierre GOUTON, for examining this thesis.

I would like to thank Ms. Lijun WEI, thank you very much for your encourage and help in the early stage of PhD study. Also Thanks very much for Mr. You LI, Mr. Hongjian WANG and Mr. Julian MURGIA, for their help in my study and research. I would also like to thank my colleagues who helped me in all these years of my thesis: Mr. Yong FANG, Mr. Tao YANG, Ms. Citlalli GAMEZ and all the others who worked with me. It is really happy and helpful to communicate and worked with them.

I also express my gratitude to Prof. Fadi DORNAIKA, Professor at the University of the Basque Country (Spain), whose advice is always valuable.

Lastly, I would like to express a special thanks to my family and friends, thanks for their emotional supports and research help.

Contents

1 Introduction **7**

1.1 Background 7

1.2 Place Recognition Based Visual Localization 9

1.3 Problem Statements and Objectives 10

1.4 Experimental Platform 13

1.5 Thesis Organization 14

2 Related Works on Place Recognition Based Visual Localization **17**

2.1 Overview 17

2.2 Approaches Based on Different Place Describing Features 20

2.2.1 Methods based on local features 21

2.2.2 Methods based on global features 23

2.2.3 Methods based on local and global features combination 25

2.2.4 Methods using three-dimensional information 27

2.2.5 Methods using deep learning features 29

2.3 Approaches Based on Place Recognizing Methods 31

2.3.1 Methods based on single image matching 31

2.3.2 Methods based on sequence matching 33

2.4 Conclusions 34

3 Visual Localization by Place Recognition Based on Multi-feature (D-λLBP++HOG) **37**

3.1 Introduction 37

3.2 Overview of used Image Descriptors 41

3.2.1 LBP (Local Binary Pattern) . . . 41
3.2.2 CLBP (Complete Local Binary Pattern) . . . 43
3.2.3 CSLBP (Center-symmetric local binary patterns) . . . 45
3.2.4 CSLDP (Center-Symmetric Local Derivative Pattern) . . . 46
3.2.5 XCSLBP (extended CSLBP) . . . 47
3.2.6 HOG (Histograms of Oriented Gradients) . . . 47
3.3 Overview of Proposed Approach . . . 48
3.3.1 Image preprocessing . . . 50
3.3.2 Block based feature extraction . . . 51
3.3.3 Multi-feature concatenation . . . 54
3.3.4 Feature comparison and image matching . . . 54
3.3.5 Final matching validation . . . 55
3.3.6 Visual localization . . . 55
3.3.7 Algorithm of multi-feature based visual localization . . . 56
3.4 Experimental Setup . . . 56
3.4.1 Datasets and ground-truth . . . 56
3.4.2 Image preprocessing and feature extraction . . . 58
3.4.3 Performance evaluation . . . 60
3.5 Experiments and Results . . . 61
3.5.1 Comparison of the different binary features and image block sizes . 61
3.5.2 Performance of multi-feature combination . . . 64
3.5.3 LSH based visual recognition . . . 66
3.5.4 Visual localization results . . . 69
3.6 Conclusion and Future Works . . . 73

4 Visual Localization Across Seasons Using Sequence Matching Based on Feature Combination 75

4.1 Introduction . . . 76
4.2 Proposed Visual Localization Approach . . . 78
4.2.1 Image preprocessing . . . 79

4.2.2 Feature extraction . . . 79
4.2.3 Image sequence matching . . . 83
4.2.4 Matching validation . . . 84
4.2.5 Visual localization . . . 84
4.2.6 Algorithm of proposed visual localization . . . 84
4.3 Experimental Setup . . . 86
4.3.1 Dataset and ground-truth . . . 86
4.3.2 Evaluation method . . . 88
4.4 Experiments and Results . . . 88
4.4.1 Feature combination analysis . . . 88
4.4.2 Sequence length selection . . . 89
4.4.3 Visual localization under different season couples . . . 92
4.5 Conclusion and Future Works . . . 97

5 All-environment Visual Localization Based on ConvNet Features and Localized Sequence Matching 99

5.1 Introduction . . . 99
5.2 Proposed Approach . . . 102
5.2.1 ConvNet features extraction . . . 104
5.2.2 Feature comparison . . . 107
5.2.3 Localized sequence matching . . . 107
5.2.4 Final matching validation . . . 110
5.2.5 Visual localization . . . 110
5.2.6 Algorithm of proposed visual localization . . . 110
5.3 Experimental Setup . . . 112
5.3.1 Datasets and ground-truth . . . 112
5.3.2 Performance evaluation . . . 115
5.4 Experiments and Results . . . 116
5.4.1 Performance comparison between single image and sequence based approach . . . 116

5.4.2 ConvNet features layer-by-layer study 117
5.4.3 LSH based visual recognition 123
5.4.4 Visual localization results 125
5.5 Conclusion and Future Works 127

6 Conclusion and Future Works **129**
6.1 Conclusion 129
6.2 Future Works 130

Chapter 1

Introduction

The overall aim of this thesis is to improve place recognition for mobile robots or vehicles visual localization in changing environments. This thesis presents methods allowing to increase the robustness of visual localization system by improving the place recognition performance using appearance rather than metric information. Our approach allows robot or vehicle to globally re-localize itself by recognizing places it has previously visited under variations in appearance and illumination.

This chapter introduces the main topic of this thesis that is place recognition based visual localization, and the effect of environmental change on place recognition reliability. It begins by outlining the background of the research topic (Section 1.1). Section 1.2 introduces the advantages of place recognition based visual localization using camera. Section 1.3 illustrates the problems caused by changing environments as well as the research objectives and contributions of this thesis. Section 1.4 describes the experimental platform used to acquire datasets for evaluating the proposed approaches. The thesis organization is outlined in Section 1.5.

1.1 Background

During the past decades, autonomous mobile robots and intelligent vehicles (IV) have obtained increasingly attention and developments from research society and industry community [12, 19]. In order to promote the development of autonomous vehicles, American

Department of Defense has hold autonomous vehicle competition called DARPA (Defense Advanced Research Projects Agency) Grand Challenge. This challenge attracted many top-level research institutes (See Fig.1-1). In 2009, Google started self-driving car project, it has self-driven for more than 1.5 million miles and is currently out on the streets of Mountain View and so on. In 2014, google released its new version of self-driving car. Fig.1-2 shows the new prototype of googles driverless cars. In recent years, some companies like Baidu, BMW and Uber also launched their self-driving cars.

(a) MIT Land Rover LR3 [1]

(b) The Stanford Racing Team [2]

Figure 1-1: Example of participants in 2007 Grand Challenge

Figure 1-2: Google prototype driverless car. [3]

[1] http://grandchallenge.mit.edu/images/all
[2] http://cs.stanford.edu/group/roadrunner/photos.html
[3] https://www.google.com/selfdrivingcar

Some experimental projects in indoor environments have shown that robots can run autonomously [71]. However, long-term autonomous navigation in changing outdoor environments is still an ongoing challenge.

Autonomous cars or robots comprise fundamental systems, such as surrounding perception, navigation, driving control and localization, which make sure vehicle to be driven safely in complex environment. Among these tasks, localization is a prerequisite for accomplishing other tasks, as it determines the vehicle position in the environment [62]. Based on localization information, we can compute the position of an obstacle and perform path planning. Therefore, the correctness and accuracy of the localization system impact the functionality of an autonomous vehicle.

Visual localization can be achieved in particular through place recognition, which is the process of identifying a previously-seen location in an environment. This thesis focuses on place recognition based visual localization task. Robust visual localization system should benefit from improved performance of place recognition.

1.2 Place Recognition Based Visual Localization

In general, Global Positioning System (GPS) is commonly used for outdoor localization. GPS seems offer a simple and low-cost solution, but it requires line-of-sight satellites. GPS-denied environments, accuracy decreasing or intermittent available conditions occur frequently in areas where there are tall buildings and trees or half-outdoor space [109]. At the same time, visual sensors recently have become the primary components of many state-of-the-art place recognition and Simultaneous Localisation And Mapping (SLAM) systems [32, 70, 77, 85, 92].

There are several advantages of using camera sensors for visual localization: (1) Firstly, digital cameras are light-weight and cheap, benefit from small form factor, have modest power requirements and their size expands their applicability to smaller hardware and mass-production; (2) Furthermore, the rich image data received from cameras offers rich appearance and texture information, as well as high potential for semantic interpretation; (3) Finally, a camera can provide information about far away landmarks. Camera such

as bumblebee2 can see large landmarks (such as mountains) hundreds of meters away. In contrast, LiDAR such as the Velodyne HDL-64E [4] has maximum range of 120m and costs higher than camera.

Vision localization using camera is often based on place (image) recognition to locate position within the world, which typically adopts image or sequence matching (retrieval) method [24,29,46] to recognize a previously-seen place in an environment. Here, the place may either be considered as a precise position— "a place describes part of the environment as a zero-dimensional point" , or as a larger area— "a place may also be defined as the abstraction of a region" where a region represents a two-dimensional subset of the environment [55]. According to this, each place can be represented by an image or sequence, thus visual localization can be achieved through place recognition based on image or sequence matching (retrieval) technique [35].

In place recognition based localization system, as Fig.1-3 shows, representative images are captured from the environment and stored in a database, with their corresponding location (GPS information). During on-line localization, each observation (image) is compared to the images of database. The location whose corresponding image (from the database) best matches to the current observation is then considered to be the currently visible location (process of place recognition). Then, visual localization is realized through the GPS information from the matched image. Recently, visual localization has been largely facilitated thanks to the progress of image features. Effectively, feature extraction enables efficient describing of environments. Thus, matching current visual input with a set of images of known places can be conveniently conducted based on extracted features.

1.3 Problem Statements and Objectives

Visual localization based on place recognition is a well-defined but extremely challenging problem to solve especially in complex outdoor environments. Most of existing place recognition based localization systems can perform successfully in static environments but fail in highly dynamic environments. In particular, visual localization systems are sus-

[4] http://velodyneLIDAR.com/hdl-64e.html

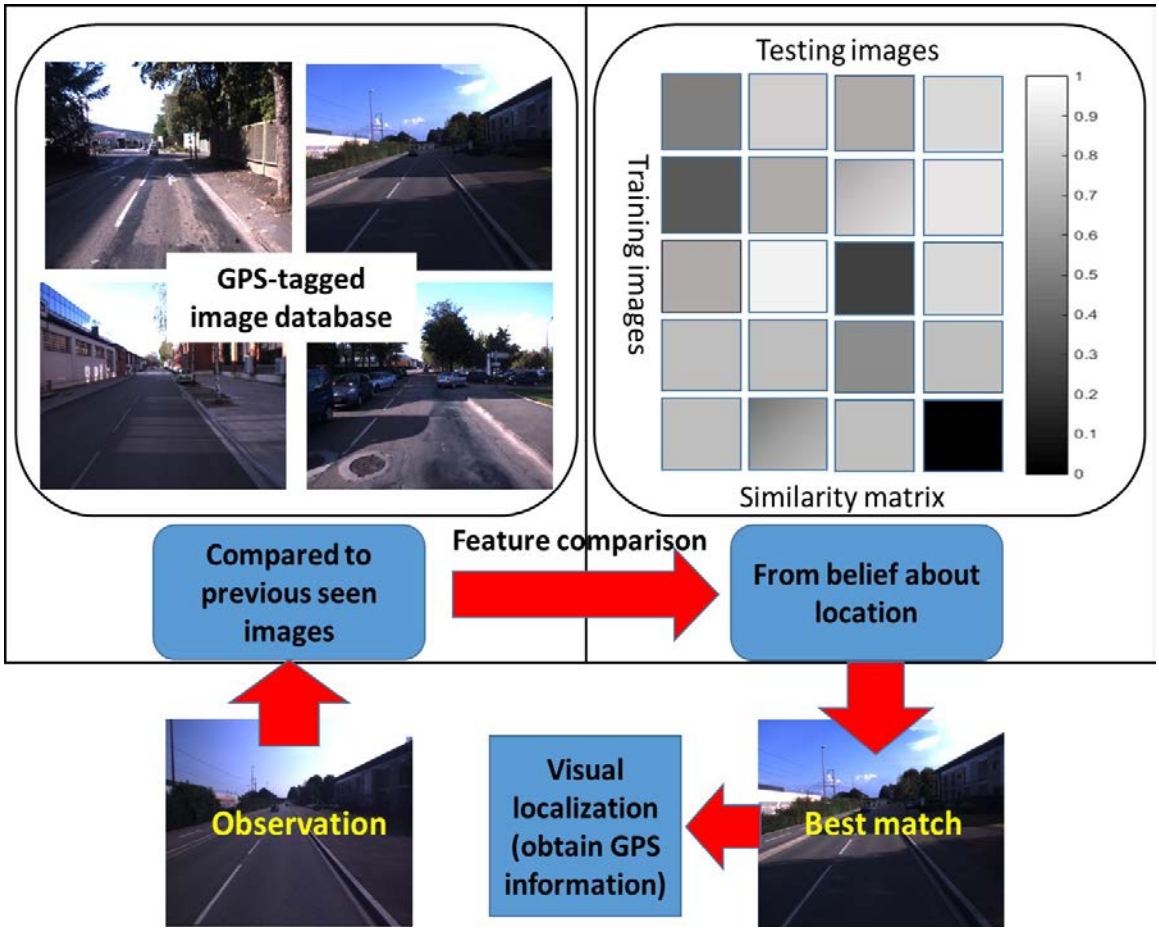


Figure 1-3: Synopsis of place recognition based visual localization system.

ceptible in large-scale changing environments where drastic appearance and illumination changes, caused by weather conditions or seasonal changing occur. This thesis addresses the challenge of improving place recognition techniques so that global perceptual changes in the environment do not cause complete failure of the robot or vehicle localization system.

Given two observations (typically images), judging whether these observations are collected from the same location will meet several troubles. Firstly, the biggest challenge is how to describe a place without being affected by environment changing. Two images from the same place may look different due to variances in appearance or illumination. In order to perform robust localization task using vision, it is necessary to describe the acquired images (or sequences) and to be able to compare these descriptions. Consequently, the recognition performance and the localization results will directly rely on the method used for visually describing the different environment locations. In addition, among local or

global features, how to combine advantageous of these features for place describing is also a challenge. Secondly, recognizing or matching methods still can be improved. The appropriate recognizing or matching strategy can improve the recognition accuracy. There are two main place recognizing methods—based on image matching and —based on sequence matching. In general, sequence matching is more robust. The last challenge is computation time. The robots or intelligent vehicles need to localize themselves in high speed driving situation, therefore the judging method for visual localization system should performed fast enough to satisfy the real-time requirement.

Our research is part of the project CPER "Intelligence du Véhicule Terrestre" (Intelligence of ground vehicle), conducted within IRTES-SET in UTBM. The goal of this thesis is to improve place recognition performance for visual localization in changing environment. The proposed approaches are tested on extensive outdoor environment datasets (some datasets are acquired by our own vehicle platform and others are public datasets) and the final outcome of this thesis is an all-environment visual localization system that is capable of running in real-time.

The three main objectives and contributions are explained as follows:

(1) The first objective is to explore feature combination for vehicle localization. Since different types of feature have their own advantages, combining some powerful features will be helpful for place recognition. A new multi-feature (D-CSLBP++HOG) is proposed for visual localization. D-CSLBP++HOG feature combine HOG (Histogram of Oriented Gradients) and CSLBP (Center-symmetric local binary patterns) features that are built from both gray-scale image and disparity map. The integration of disparity information, permits to improve the performance of place recognition, especially in complex environment situation. In addition, for real-time visual localization, local sensitive hashing method (LSH) is used to compress the high dimension of the multi-feature into binary vector. It can thus speed up the process of image retrieval. To show its effectiveness, the proposed method is tested and evaluated using real datasets acquired in outdoor environments. As we will show, our approach allows more effective visual localization compared with the state-of-the-art FAB-MAP (Fast Appearance Based Mapping) method.

(2) When single image matching is used for visual localization, it is easy to perceive two

different places as the same due to seasonal changing. This is known as "perceptual aliasing". In order to decrease the perceptual aliasing influence, retrieval based on sequence of images rather than on single image can be used to improve place recognition results. Therefore, another task is to develop a visual localization system based on sequence matching to improve the recognition performance and localization results. An approach of visual localization across seasons is proposed using sequence matching based on the combination of GIST and CSLBP features. Studies of the relationship between image sequence length and sequence matching performance is conducted. To show its effectiveness, the proposed method is tested and evaluated in four seasons outdoor environments. The results have shown the improved precision-recall performance against the state-of-the-art SeqSLAM (Sequence Simultaneous Localization and Mapping) algorithm.

(3) Most of the used features are hand-crafted features and they have demonstrated good performance in place recognition and visual localization. However, for database with specific surroundings (i.e. trees, buildings or mountains), it is difficult to decide what kind of features should be taken to describe places. Suitable features can achieve good place recognition results while unreliable features could lead to false recognizing. With the rapidly development of deep learning networks, it is becoming apparent in place recognition tasks that hand-crafted features are being outperformed by learnt features. Our contribution is to use the automatic learned Convolutional Network (ConvNet) features to accomplish all-environment visual localization task under appearance and illumination changing situations. A comprehensive performance comparison of different ConvNet layers is conducted on four real world datasets. To speed up the computational efficiency, locality sensitive hashing method is taken to achieve real-time performance with minimal accuracy degradation.

1.4 Experimental Platform

As illustrated in Fig.1-4, our experimental GEM vehicle is equipped with many sensors (stereo vision system, camera, RTK-GPS, etc). A Bumblebee XB3 stereo vision system is installed on the top, and oriented to the front. It is composed of three collinear cameras

Figure 1-4: Experimental vehicle equipped with sensors (especially camera and RTK-GPS).

with a maximum baseline of 240 mm. Images are captured in a format of 1280×960 pixels. In our application, only the left and right cameras are used. A RTK-GPS receiver (10 Hz) is used to collect position (GPS) information. All the installed sensors are fixed rigidly.

1.5 Thesis Organization

The rest of the thesis is divided into five chapters:

In chapter 2, existing approaches for place recognition based visual localization are reviewed. According to the features used in place describing, the visual localization methods are classified as: approaches based on global descriptors, approaches based on local features, approaches based on multi-feature combination, approaches based on 3D information, approaches using deep learning features. According to the place recognizing method, two main approaches are introduced: image matching/retrieval approach and sequence matching/retrieval approach.

In chapter 3, a visual localization method based on multi-feature combination and disparity information using stereo camera is proposed. Disparity information is integrated into complete center-symmetric local binary pattern (CSLBP) to obtain robust global im-

age description (D-CSLBP). In order to describe the scene more accurately, multi-feature combining D-CSLBP and HOG features is adopted to provide valuable information and to decrease the effect of some typical problems in place recognition such as perceptual aliasing. It improves visual recognition performance by taking the advantage of depth, texture and shape information. In addition, for real-time visual localization, local sensitive hashing method (LSH) is used to compress the high dimensional multi-feature into low dimensional binary vector. The proposed method is tested and evaluated using real datasets acquired in outdoor environments.

In chapter 4, visual localization across seasons using sequence matching is presented. Matching places by considering sequences instead of single images denotes higher robustness to extreme perceptual changes. The recognition results of different sequence lengths is compared. The results obtained from Nordland dataset shows that the proposed method can achieve better performance than SeqSLAM method.

In chapter 5, all-environment visual localization system based on ConvNet features and localized sequence matching is proposed. The pre-trained network provided by MatConvNet is used to extract features and then a localized sequence matching technique is applied for visual recognition. Compared with the traditional approaches based on hand-craft features and single image matching, the proposed method shows better performances even in presence of appearance and illumination changes. A comprehensive performance comparison of different ConvNet layers (each defining a level of features) is conducted considering both appearance and illumination changes.

In chapter 6, a summary of the achieved outcomes and discussions of their relevance to the current research in place recognition based visual localization are provided. In addition, some research perspectives for future work are also indicated.

Chapter 2

Related Works on Place Recognition Based Visual Localization

This chapter presents an overview of relevant works in the field of place recognition based visual localization. It begins by defining the core aspects of place recognition based visual localization, namely: place describing, place remembering and place recognizing. Then, relevant works about place describing and place remembering are summarized respectively.

2.1 Overview

With the low cost of cameras and the richness of provided sensor data , place recognition for visual localization is attracting more and more attention [21,23,78,103]. In this context, each place (location) can be represented by an image or sequence of images, and a robot or vehicle localizes itself by identifying a previously-visited location through image or sequence retrieval. Thus, place recognition based approaches have to be robust even in situations in which the robot's metric position estimation is largely erroneous.

Solving the all-environment visual localization problem for identifying where a robot is over time has become one of the main challenging research areas [47]. Unfortunately, this is not an easy task, because place appearance strongly changes at different times of day, along months and especially along seasons [6, 100].

Fig.2-1 presents a general scheme of a place recognition based visual localization sys-

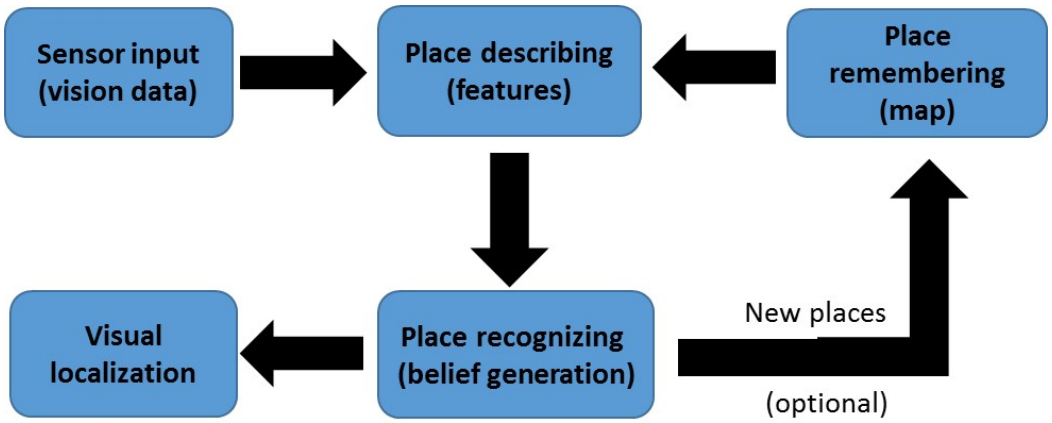


Figure 2-1: General scheme of visual place recognition system, consisting of five core components. Incoming visual data is processed by place describing module. Robot's knowledge of the world is stored in place remembering module. Place recognizing module decides whether the current visual data matches a previously stored place. Throughout the place recognition process, robot localizes itself thanks to matching with a previously-visited location. Since the previously-visited location tagged with GPS information, the vehicle or robot can achieve its localization by assimilating its position with the one of the retrieved/matched image.

tem. This localization system has the following essential components:

(1) Visual information inputs: Images or videos are the data source for the whole system. It also includes data preprocessing, which transforms observations into a suitable form for description or storage, e.g. collection of feature descriptors or whole images.

(2) Place describing: Places must be described in a way that enables them to be efficiently stored and recognized when they are revisited. Visual place description techniques mainly fall into two broad categories: those that selectively extract parts of the image that are in some way interesting or notable; and those that describe the whole scene, without a selection phase.

(3) Place remembering: A place recognition system needs to refer to a map, where the extracted descriptors are stored and organized for comparison and retrieval. The state-of-the-art approaches can be divided into three main categories: those using place databases, those using topological maps, and those using metric maps. The appropriate type of place remembering depends on the purpose of the place recognition system.

- A place database is the simplest mean to represent a particular environment where

only appearance information is stored. In this approach, place recognition is based solely on appearance similarity and applies image retrieval techniques. Although valuable information is lost due to not including of relative pose information, there are computationally efficient indexing techniques that can be exploited.

- Topological maps contain relative information about places in an environment. It can simply consists of an ordered collection of images: a linear database that reflects the order in which places are consistently encountered when an environment is visited. In these cases, localization simply identifies the most likely location.

- Metric maps depict the absolute scale of environments more accurately, maintaining a lot of information about environment details, such as distances, driving direction or landmark position, and they are usually referenced according to a global coordinate system. This representation is most appropriate for vehicle localization and guidance. However, metric maps are more difficult to build and maintain, and are computationally demanding.

(4) Place recognizing: It refers to the mean of comparing current observation to the stored ones. Ultimately, the purpose of place recognition is to determine whether a place has been seen before. There is a general understanding that if two place descriptions appear similar there is a greater likelihood that they have been captured at the same physical location. Thus, the central goal of any place recognition system is reconciling visual input with the stored data to generate a belief distribution. There are two main approaches for place recognizing: based on single image or based on sequence of images. Sequence-based approach is more robust by removing some false positive recognition.

(5) Visual localization: Throughout the place recognition process, robot localizes itself based on a previously-visited location, which are tagged with GPS information. The vehicle or robot can then achieve its localization by assimilating its position to that of the retrieved image through the recognized place. Using place recognition based visual localization, accumulation error that often occur in odometry-like approach can be avoided. It also should be noted that, the system simply identifies the most likely places and then get a rough position. This is a topological level localization rather than a very accurate metric

localization.

Among the module of a place recognition based system, place describing and remembering are the two cores. In order to perform visual localization using vision, it is necessary to describe the acquired images and to be able to compare their descriptions. Consequently, the quality of place remembering and the posterior localization will directly rely on the method used for visually describing the different environment locations.

Due to the above reasons, we classify the different visual localization approaches according to the description method employed as: approaches based on local features, approaches based on global features, approaches based on multi-feature combination, approaches using three-dimensional information and approaches using deep learning features.

On the other hand, from the perspective of recognizing method, visual localization methods can also be divided into two mainly categories: methods based on single image matching; and methods based on sequence matching.

In this chapter, we review the main approaches with regard to place recognition based visual localization and SLAM. The rest of this chapter is organized as follows: Section 2.2 enumerates fundamental works based on place describing features. Section 2.3 introduces two main place recognition based visual localization approaches: approaches based on single image matching and approaches based on sequence matching. Section 2.4 concludes the chapter, and proposes some open research lines.

2.2 Approaches Based on Different Place Describing Features

Each place is a unique location and it must be described in a way that enables it to be efficiently stored and recognized when revisited. Many issues relevant to place recognition-based approaches have been proposed. The different methods can be classified according to the way of place description employed as: methods that describe places by local features; methods that describe places by global features; methods that describe places by local and global features combination; methods that describe places using three-dimensional

information and methods that describe places using deep learning features.

2.2.1 Methods based on local features

Local feature methods make use of distinct features (or feature keypoints) within images, which capture the essence of the image [13, 114]. During extraction step, a set of distinct features (or feature keypoints, i.e. corners or edges) is first detected by analyzing images and searching for distinctive pixel patterns (see Fig.2-2). Then, a description step is performed, where some measurements (i.e. comparison or concatenation) are taken from the vicinity of each local descriptor to form the final feature [35]. In general, features are formed as a multi-dimensional floating-point vectors or bit strings.

A good local feature typically needs to be invariant to one or more affine transformations—such as image scale and camera rotations. Thus, the same local features can be identified in the similar images, which enables recognition of familiar places. Many works have used local features in the field of place recognition and visual detection tasks [37], especially since the development of Scale-Invariant Feature Transform (SIFT) algorithm [63].

Table 2.1: Summary of main local features

Name	Dimensions	type	References
SIFT	128	Float	[63]
SURF	32, 64, 128	Float	[9]
PCA-SIFT	36	Float	[49]
KAZE	64	Float	[3]
LBP	59	Bit	[80]
CLBP	514	Bit	[42]
CSLBP	16	Bit	[43]
CSLDP	16	Bit	[112]
XCSLBP	16	Bit	[95]
BRISK	512	Bit	[61]
BRIEF	512	Bit	[16]
ORB	256	Bit	[90]
FREAK	512	Bit	[1]
AKAZE	488	Bit	[2]
LDB	256, 512	Bit	[6]

Table 2.1 collects some of the main local features used in place recognition. For ex-

ample, Murillo et al. [9] use SURF while FrameSLAM [52] adopts center-surround feature (CenSurE). Kawewong et al. [48] use Position-Invariant Robust Features (PIRFs): each place is described by a dictionary of these representative PIRFs, whose appearance variation is assumed relatively small with regard to robot motion. Andreasson and Duckett [4] present a simplified version of the SIFT algorithm—M-SIFT (Modified SIFT features) which selects interest points from omni-directional images. The results show that the method based on local features M-SIFT obtain a significant high level of performance and robustness to environmental variations.

Recently, a number of local binary features have been proposed in the literature, providing an interesting research line to explore for place description and recognition [98]. Their advantages are that they are invariant to monotonic changes in gray-scale and fast to calculate. One typical binary feature LBP (Local Binary Pattern) [80] is used in paper [84], where SVM (support vector machine) recognition models are built based on the extracted

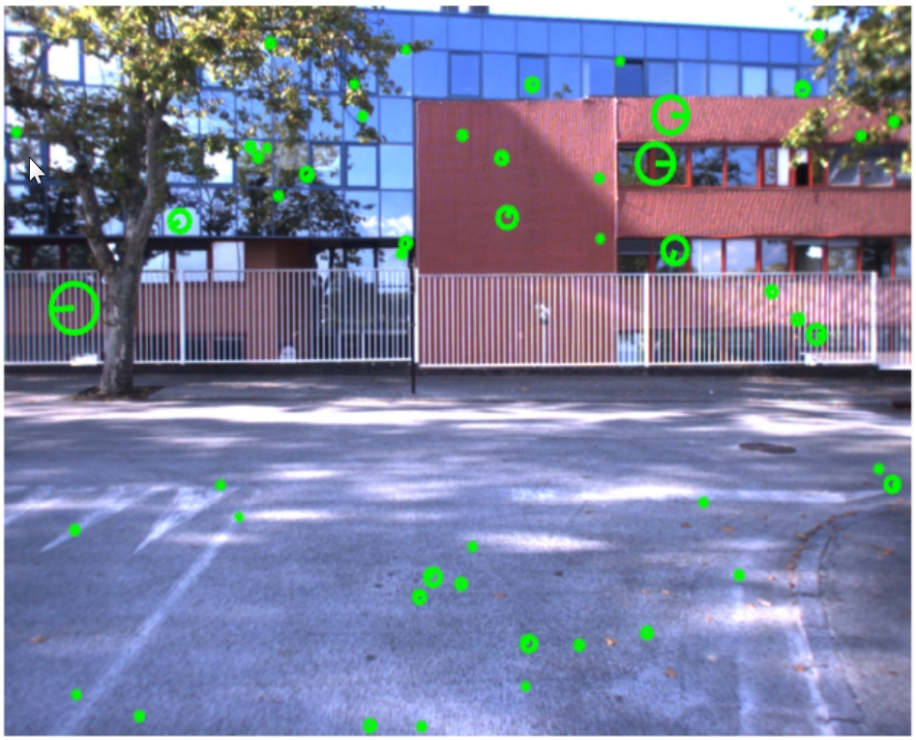


Figure 2-2: SIFT extracts interest points in an image for description. The circles are interest points selected by SIFT within the image. The number of possible features may vary depending on the number of interest points detected in the image.

LBP features and each place can be recognized using these SVM models.

Other local features based on keypoints detection like BRIEF (Binary Robust Independent Elementary Features) [16], ORB(oriented BRIEF) [90], BRISK (Binary Robust Invariant Scalable Keypoints) [61], Local Difference Binary (LDB) [6], FREAK (Fast Retina Keypoints) [1], BRIEF (Binary Robust Independent Elementary Features) [16] and KAZE [3] are also used in place recognition.

Local features present high discrimination capacity, resulting into higher recognition rates and less detection errors. However, the total local features dimension of each image could be very high, and directly matching image features can be inefficient. The bag-of-words model [96] increases efficiency by quantizing local features into a vocabulary, where every feature is assigned to a particular word and each image can be described by low dimensional vector or binary string. As the bag-of-words model ignores the geometric structure of the place that is described, the resulting place description is pose invariant; that is, the place can be recognized regardless of the position of the robot with respect to the place.

The biggest disadvantage of local image features is that they perform poorly—or fail entirely—in the presence of extreme condition variance [39]. In such cases, either feature detection and matching process fail because the object of interest being described by local descriptors is less distinctive in different conditions.

2.2.2 Methods based on global features

In the previous section, solutions based on local features were reviewed, where the description pays more attention to parts of image (sub-regions). Such descriptions work well for partial occlusions or camera rotations, but are not able to deal with general structure or framework of the whole scene.

Global features describe the image in a holistic manner, using the whole-image (or global descriptors) rather than (sub-regions) for place recognition. Global features are normally very fast to extract and are more robust to the environment changing conditions where particular features are unrecognizable [35]. Some of the main global features used

in place recognition and scene classification approaches are shown in Table 2.2. Exam-

Table 2.2: Summary of main global features

Name	References
WI-SIFT	[8]
WI-SURF	[8]
Gradient Orientation Histograms	[53]
Principal Components	[36]
Colour Histograms	[103]
GIST	[81]
BRIEF-GIST	[99]
DIRD	[59]

ples of some global features used in early visual localization systems include: color histograms [103], features based on principal component analysis [36] and Gradient Orientation Histograms [53]. In paper [103], the authors describe place using six one-dimensional color histograms from HLS and RGB color spaces. The reference images are retrieved using a nearest neighbor learning scheme in their topological map and they obtains at least 87% of correctly classified images in their whole appearance-based place recognition system.

Kosecka et al. [53] propose a vision-based navigation strategy using gradient orientation histograms as image feature. The similar places are determined through comparison of these gradient orientation histograms.

Winters et al. [110] utilize an omni-directional camera to create a topological map. The large image set is compressed using PCA to form a low dimensional eigenspace, then robot could determine its global topological position using an appearance based matching method.

Besides that, one of the popular global features—GIST [81], which was initially developed for scene recognition, has already been used for place recognition on a number of research works [99]. GIST uses Gabor filters at different orientations and different frequencies to extract information from the image. The results are averaged to generate a compact vector that represents the "GIST" descriptor of the scene.

Global features can be also constructed by detecting and concatenating local features.

For example, Lamon et al. [57] propose fingerprints, composed of a variety of image features –such as color patches, edges and corners. By ordering these features in a sequence between 0° and 360°, place recognition could be reduced to string-matching. These systems used omni-directional cameras which allow rotation-invariant matching at each place.

Alternatively, Badino et al. [8] apply SURF descriptor to entire images, constructing a single feature Whole-Image SURF (WI-SURF) for each place. The successful results for long-term localization experiments are reported, concluding its validity for solving the global localization problem. Similarly, motivated by the success of GIST and BRIEF binary descriptors, BRIEF-GIST [99] is proposed. BRIEF feature is computed in a similar whole-image fashion [16]. BRIEF-GIST demonstrated high computational efficiency and no requirement for vocabulary training is needed.

Other possible implementation consists in partitioning the image into a grid, computing descriptor for each patch and concatenating the obtained descriptors to form the final feature. In paper [59], an illumination robust feature (DIRD) is proposed based on normalized filter responses of small images regions. The experiments showed that DIRD achieved good performance for loop closure detection. Arroyo et al. [7] divide each panorama into sub-panoramas and extract LDB (Local Difference Binary) binary descriptor for each sub-panorama. The final image feature is created by concatenating the different LDB descriptors. The proposed LDB feature also achieved good performance in life-long visual localization.

In general, global image features are easier to compute and save storage space, they are better suited to varying conditions and can be modified into more robust patch-normalized or shadow-invariant forms, they are more sensitive to camera viewpoint. Some practical viewpoint problems—such as those due to vehicle rotation—can be ameliorated with panoramic imagery and the modified GIST descriptor.

2.2.3 Methods based on local and global features combination

Global and local features demonstrate useful interest for place recognition based visual localization. However, each has its own advantages and disadvantages. In order to maximize

the benefits of each feature type, several authors have proposed solutions based on combination of different image features for place recognition based visual localization [47].

Murillo et al. [75] propose a three-step hierarchical localization method using omni-directional images. A global color descriptor is applied to obtain a set of susceptible loop candidates, and then line features described by their line support regions are matched using pyramidal matching in order to find the most similar image given a predefined visual memory.

Another approach is to use a global descriptor to perform a fast selection of similar images during image searching and then use a more accurate process to confirm the association, such as matching local features. Goedemé, et al. [40] present a localization system using omni-directional cameras where, for each acquired image, vertical column segments are extracted and described with ten different descriptors. After a clustering process, these local descriptors are inserted into a kd-tree structure that is used by the localization process. When a query image arrives, the same local descriptors applied to the vertical structures are computed over the entire image and used to rapidly retrieve possible loop candidates. Next, a matching distance based on the column segments is applied between the image and each of the candidates in order to ensure a correct image matching.

In work [65], a robust and real-time visual place recognition algorithm is proposed by combining the local visual features FAST (Features from Accelerated Segment Test) and CSLBP (Complete Center-symmetric Local Binary Patterns). Based on the proposed features, bag-of-features and support vector machines are used to realize place recognition based on omni-directional vision for mobile robots. The experimental results show that the robot can achieve robust place recognition with high classification rate in real-time.

Wang and Lin present a combined local and global descriptor for omni-directional images called Hull Census Transform (HCT) [107], which consists of repeatedly generating the convex hull from the extracted SURF features and computing the relative magnitude between these features that compose the convex hull, resulting into a set of binary vectors. This representation is then used for detecting scene changes.

A location recognition system which combined edges, local features and color histograms was proposed by Wang and Yagi [106]. In this system, image description process

is computed in an integrated way: the Harris detector is used to obtain both edges and interests points, while SIFT is used for describing interest points.

2.2.4 Methods using three-dimensional information

In addition to description of the places with 2D model directly in the visual domain (instead of making a geometric-model), they can also be extended with metric information [76]. Thus, 2D image with metric information can be regarded as three-dimensional (3D) information. Metric range information can be inferred using stereo cameras [27]. Monocular cameras can also infer metric information using structure-from-motion algorithms as in the following methods: MonoSLAM [28], PTAM [50], LSD-SLAM [17], and ORB-SLAM [74].

Several works in the literature use 3D (three-Dimensional) information to improve performance of place recognition based localization methods. In [24], FAB-MAP (Fast Appearance Based Mapping) is extended to incorporate the spatial distribution of visual words in 3D. Similarly, a combination of visual words with 3D information from stereo sequences is used in [15] to perform robust place recognition.

Morioka et al. [73] propose a SLAM navigation method that is effective even in crowded environments by extracting robust 3D PIRF (Position Invariant Robust Feature) points from sequential images and odometry.

In paper [34], the authors present a variant of SURE, an interest point detector and descriptor for 3D point clouds and depth images, and use them for recognizing semantically distinct places in indoor environments. They also demonstrated that SURE features are well suited for place recognition using a bag-of-words approach.

Paper [68] describes a new system CAT-SLAM (Continuous Appearance based Trajectory SLAM), which augments sequential appearance-based place recognition with local metric pose filtering to improve the frequency and reliability of appearance based loop closure. An extension of CAT-SLAM called CAT-Graph is introduced in [67], combining visual appearance and local odometry data as in CAT-SLAM, but fuses multiple visits to the same place into a topological graph-based representation of indoor environments. It

demonstrates that loop closure detection in a large urban environment with capped computation time and memory requirements and performance exceeding FAB-MAP by a factor of 3 at 100% precision

Cadena et al. [14] introduce a place recognition framework based on stereo vision which combined a bag-of-word model for obtaining loop closure candidates and an algorithm based on CRF-Matching (Conditional Random Fields-Matching) in order to verify these candidates. This matching method is more robust than using only epipolar geometry, since it used 3D information provided by the stereo images.

Paper [15] proposes a place recognition algorithm for SLAM system using cameras. It considers both appearance and geometric information of interest points in the images. Hypotheses about loop closings are generated using a fast appearance-only technique based on the bag-of-words method. According to the indoor and outdoor data experiments, it shows that the proposed system can attain at least full precision (no false positives) for high recall (fewer false negatives).

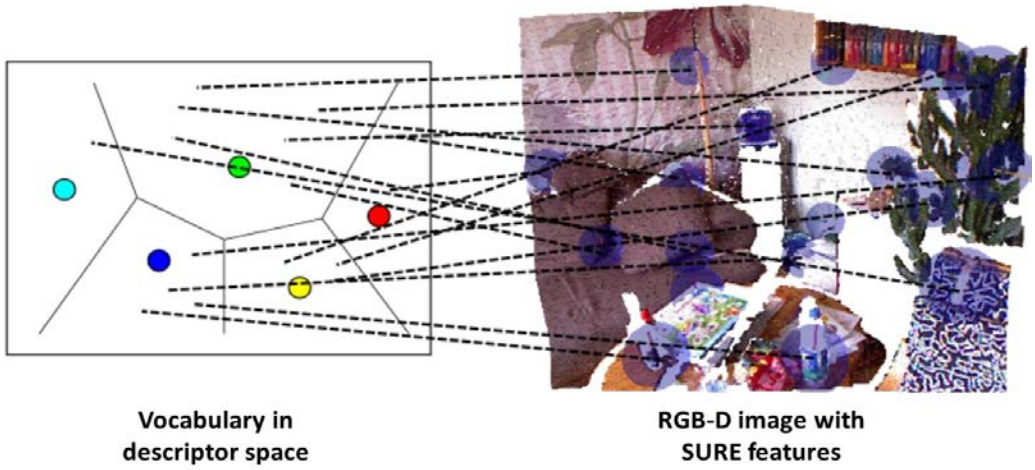


Figure 2-3: An example of detecting SURE features in depth images at locations with locally prominent surface curvature (from [34]). SURE feature captures local shape and colored texture at interest points. Based on SURE features, places are recognized using a bag-of-words approach.

Many other systems use data from additional sensors such as RGB-D cameras [31]. These sensors provide dense depth information as well as image data and then can exploit 3D object information to improve place recognition and localization accuracy [33]. As

illustrated in Fig.2-3, Torsten Fiolka et al. [34] present a variant of SURE—an interest point detector and descriptor for 3D point clouds and depth images. The SURE operator selects distinctive points on surfaces by measuring the variation in surface orientation based on surface normals in the local vicinity of a point. Furthermore SURE includes a view-pose-invariant descriptor that captures local surface properties and incorporates colored texture information. The experiment results demonstrate that SURE features are well suited for place recognition in some simple environments.

2.2.5 Methods using deep learning features

Place recognition methods based on the hand-crafted features are prone to be affected by the changing of illumination or appearance. Their performance in challenging environments strongly depends on the invariance of those descriptors to perceptual changes. Nowadays, it is rapidly becoming apparent that in recognition tasks hand-crafted features are being outperformed by deep learning features [41]. Deep learning features obtained from deep neural networks show strong power for place describing [87]. Thanks to the deep neural networks, place recognition based visual localization using automatic learned features is interesting and promising.

Convolutional Neural Network (ConvNet) as one of the popular deep neural networks, was firstly proposed by LeCun et al. [60] in 1989. ConvNet features are learned automatically from datasets through multi-layer supervised networks. ConvNets permit to achieve significant performance improvement on object classification or recognition, and outperform traditional hand-crafted features based approaches [5].

In paper [22], a place recognition technique based on ConvNets model is presented by combining the powerful features learned by ConvNets with a spatial and sequential filter. Applying the system to a 70 km benchmark place recognition dataset (Eynsham dataset), 85.7% recall is achieved at 100% precision.

Sünderhauf et al. [102] present a novel place recognition system that is built on state-of-the-art object detection methods and convolutional visual features. As illustrated in Fig.2-4, the astonishing power of convolutional neural network features is used to identify matching

Figure 2-4: Place recognition system utilizing convolutional network features as robust landmark descriptors to recognize places despite severe viewpoint and condition changes, without requiring any environment-specific training. The colored boxes in the images above show ConvNet landmarks that have been correctly matched between two significantly different viewpoints of a scene. This enabling place recognition under challenging conditions (from [102]).

landmark proposals between images to perform place recognition over extreme appearance and viewpoint variations. The experiment results have also revealed further insights: mid-level ConvNet features appear to be highly suitable as descriptors for landmarks of various sizes in a place recognition context.

Paper [101] presents a thorough investigation on the utility of ConvNet features for the important task of visual place recognition in robotics. Then, a novel method is proposed by combining the individual strengths of the high-level and mid-level feature layers to partition the search space and to recognize places under severe appearance changes. In addition, locality-sensitive hashing and novel (semantic search space partitioning) optimization techniques are used for real-time place recognition. Comprehensive study on four real world datasets highlighted that the proposed method performed better for place recognition when faced with appearance and viewpoint changes.

In paper [5], the authors develop a convolutional neural network architecture that is trainable in an end-to-end manner. The main component of this architecture is a new generalized VLAD layer (NetVLAD). The layer is readily pluggable into any convolutional neural network architecture and amenable to training via back-propagation. The proposed architecture significantly outperforms non-learnt image representations on Tokyo dataset, as well as on the Oxford and Paris image retrieval benchmarks.

In paper [41], a convolutional neural network is trained for the first time with the purpose of recognizing revisited locations under severe appearance changes. It maps images to

a low dimensional space where euclidean distance represent place dissimilarity. In order to help the network to deal with weather or illumination variations, the authors train the network with triplets of images selected from datasets which present a challenging variability in visual appearance.

In paper [20], the authors conduct a comprehensive performance comparison of the utility of features from all the ConvNet 21 layers for place recognition. In work [91], AlexNet ConvNet model was trained on the ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) for object recognition.

In addition, the availability of pre-trained network models makes ConvNets easy to experiment for place recognition. The software packages Overfeat [93], Caffe [45] and MatConvNet [105] provide network architectures pre-trained for a variety of recognition tasks. Especially, MatConvNet, an important ConvNet MATLAB toolbox designed with an emphasis on simplicity and flexibility, allows fast prototyping of new ConvNet architectures and supports efficient computation on CPU and GPU [58].

2.3 Approaches Based on Place Recognizing Methods

Place recognizing methods for visual localization can be divided into two categories: 1) methods based on single image matching; 2) methods based on sequence matching.

2.3.1 Methods based on single image matching

Traditionally, visual localization has been performed by considering place as single image. The basic technique consists of building a database of images collected off-line by a robot or a vehicle. Then the most similar to the currently acquired one can be retrieved. If two places are similar enough, they can be regarded as taken from the same location. As Fig.2-5 shows, each testing image has to retrieve its similar one from the training database. Once place is recognized based on retrieved images, the robot or vehicle can localize itself, by assimilating its position to the one of the retrieved image from the training database.

Many place recognition based visual localization approaches are realized through matching appearance of the current scene image to the training images from database [64]. FAB-

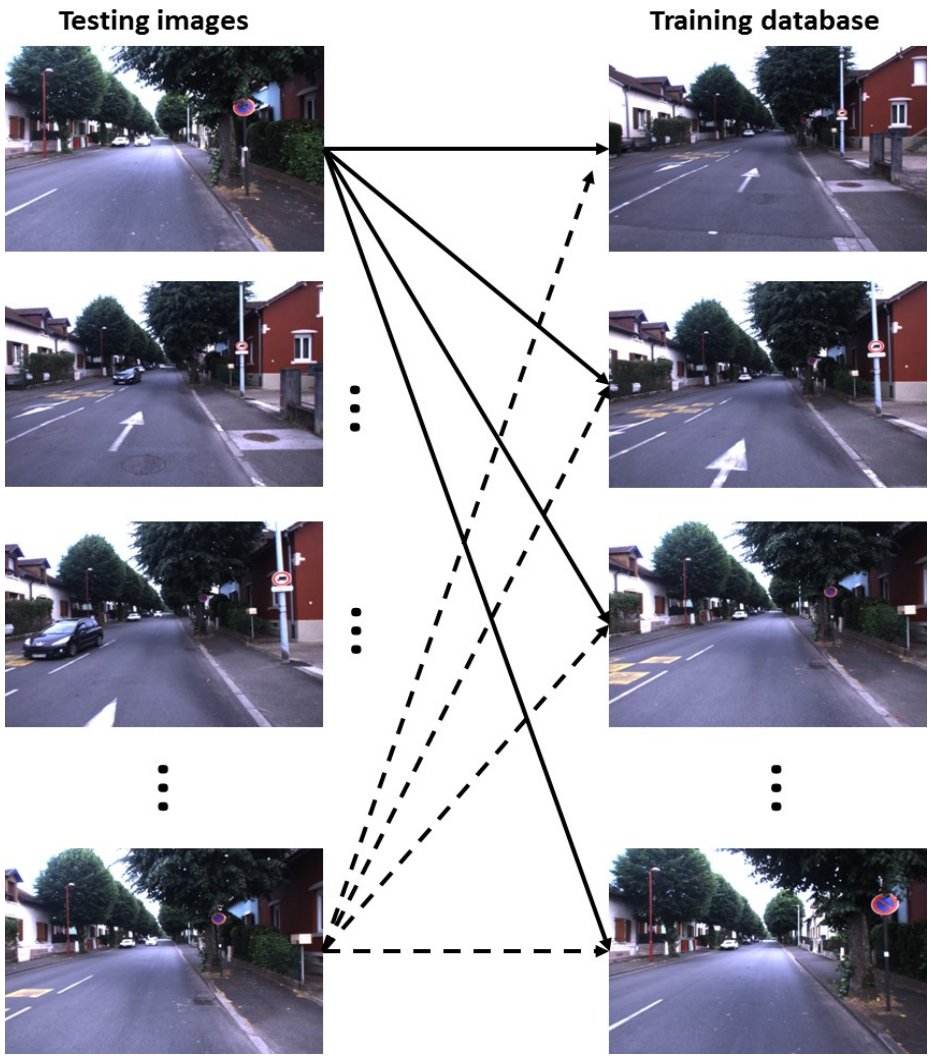


Figure 2-5: Example of place recognition based on single image. Each testing image has to retrieve its similar one from the training database.

MAP [24] can be considered as the milestone of image matching method for visual localization. It proposes to match appearance of the current scene image to a reference one by employing bag-of-words image retrieval technique. It uses a bag-of-words model with SIFT or SURF features for image description and calculates the distinctiveness of each word during a training phase. The probabilities of visual words are approximated by a Chow Liu tree, computed from a set of training data as the maximum-weight spanning tree of a directed graph of co-occurrences between visual words. FAB-MAP handles the perceptual aliasing problem by considering not only whether two locations were similar in the sense that they have many visual words in common, but also whether the words in common

were sufficiently rare so that the locations could be considered distinctive.

Knopp et al. [51] perform large-scale appearance-based localization using bag-of-feature representation, but consider only matches to individual images in the database without considering linear combination of bag-of-feature vectors.

Using single image matching for place recognition is an easy and simple way. However, when robotic systems operate in larger uncontrolled environments and for longer time periods, place recognition using single image is prone to be affected by the changing of illumination and moving objects (e.g. cars or pedestrians).

2.3.2 Methods based on sequence matching

Early place recognition systems often implicitly used the simplifying assumption that the visual appearance of each place would not change over the course of the experiment. However, as robotic systems operate in ever-larger uncontrolled environments and for longer time periods, it has rapidly become apparent that this assumption is no longer valid.

Figure 2-6: An example of place recognition based on image sequence. Sequences A and A' are taken in a time interval of two weeks.

When the appearance of an environment is changing, appearance-based place matching becomes less reliable and the relative topological structure of an environment becomes more important. Instead of calculating the single location similarity between a current image, sequences of images can be used to match places despite changes in lighting and

weather conditions, or poor visibility [72, 82]. As Fig.2-6 illustrates, sequences A and A' are taken in a time interval of two weeks, although illumination and objects (cars and trees) are changed, matching using sequence can still recognize the place successfully.

More recently, SeqSLAM (Sequence Simultaneous Localization and Mapping) [72] introduces the idea of matching places by considering sequences instead of single images like previous proposals such as FAB-MAP. In SeqSLAM method, the matrix of image similarities between local query (testing) image sequence and training image sequence is constructed firstly. Image similarity is evaluated using the sum of absolute differences between contrast enhanced, low-resolution images without the need of image keypoint extraction. The place recognition score is the maximum sum of normalized similarity scores over predefined constant velocity paths (i.e. alignments between the query sequence and database sequence images) through the matrix. Using this sequence matching approach significantly improves place recognition reliability.

The sequence-based approach can operate reliably in these conditions because it does not require the image comparison step to achieve 100% correctness — so long as the correct location is more similar than an incorrect location sufficiently frequently the sequence filter can identify the path [70].

2.4 Conclusions

This chapter presented a survey of relevant works in the field of place recognition based visual localization. Approaches of this issue have been studied extensively, however, place recognition based visual localization in changing environments can still be improved. The contributions of this thesis are motivated by the reviewed research, in particular, the presented works are inspired by multi-feature combination and sequence-based methods:

(1) As presented in section 2.2.5, feature combination and three-dimensional information can improve the performance of place recognition. Inspired by this, a new multi-feature (D-CSLBP++HOG) based visual localization will be proposed in Chapter 3. D-CSLBP++HOG combines HOG and CSLBP features that are built from both gray-scale image and disparity map. By taking advantage of texture, shape and depth information, the

proposed multi-feature is more robust for place describing, thus it can improve performance of place recognition.

(2) Considering single image matching for place recognition is fragile in drastic changing environment (i.e. different seasons), the second task is to develop sequence matching based visual localization system without being affected by seasons changing. In Chapter 4, visual localization across seasons is then proposed based on sequence matching and feature combination. Here, global feature GIST and local binary feature CSLBP are combined together for place describing. The proposed method is tested and evaluated in four seasons outdoor environments.

(3) Motivated by the success of deep learning features in visual recognition, a visual localization technique based on ConvNet networks and localized sequence matching is proposed in Chapter 5. Compared with the traditional approaches based on hand-craft features and single image matching, the proposed method can achieve good performances even in presence of appearance and illumination changes.

Chapter 3

Visual Localization by Place Recognition Based on Multi-feature (D-λLBP++HOG)

In this chapter, a multi-feature based method for vehicle visual localization in urban environments is proposed. The considered multi-feature combines HOG descriptor and D-λLBP descriptor (λLBP which is extracted from both gray-scale image and disparity map). This multi-feature takes the advantage of image texture, depth and shape information at the same time, it hence permits to achieve better place recognition performance than image single feature. To evaluate the proposed method, experiments are conducted on several real outdoor datasets. Furthermore, to speed up the process of place recognition, Locality Sensitive Hashing (LSH) is used to compress the high dimensional feature data and accelerate the process of similar images search.

3.1 Introduction

One of the prerequisites of navigation issue is to make the vehicle or robot able to reliably determine its position within its environment. With the wide use of cameras, varieties of approaches were proposed to address the challenges of place recognition based visual localization [115] [35].

As already mentioned in the literature review (Chapter 2), FAB-MAP method can be considered as the milestone in the field of visual localization. FAB-MAP approach consists in matching the appearance of current scene to a same (similar) past visited place by converting the images into bag-of-words representations built on local features such as SIFT or SURF.

In local feature based place recognition approaches, image representation is defined as collection of local features which contribute with their robustness when faced with local image variations as well as from discriminative power of their descriptors. Nevertheless, most of these works exhibit a high computation cost or complex feature extraction for image matching. Also, few works pay attention to the depth information for visual place recognition.

Recently, binary image descriptors that encode patch appearance using compact binary string with low memory requirements, are widely used in image description and visual recognition [98]. Their advantages are that they are invariant to monotonic changes in gray-scale and fast to calculate. One typical binary descriptor is LBP (Local Binary Pattern). Since it was firstly proposed in 1996, several new variants of binary descriptors have been proposed [10]. In this chapter, the most relevant binary descriptors for visual place recognition that will be tested and compared in our approach are: LBP, CLBP (Complete Local Binary Pattern) [42], CSLBP (Center-symmetric local binary patterns) [43], CSLDP (center-symmetric local derivative pattern) [112] and XCSLBP (extended CSLBP) [95]. These different local binary descriptors are noted as λLBP.

Despite local binary features efficiency, histograms of oriented gradients (HOG) features have also been successfully used in various vision tasks such as object classification, image search and scene classification [113]. Xiaoyu Wang et al. [108] combine histograms of oriented gradients (HOG) and local binary pattern (LBP), and propose a novel human detection approach capable of handling partial occlusion. For such applications, HOG is one of the best features to capture edge or local shape information which provides a rough description (shape information) of the scene.

Considering the robust and strong image representation ability of binary descriptors and HOG feature, we expect that their combination would provide more useful information and

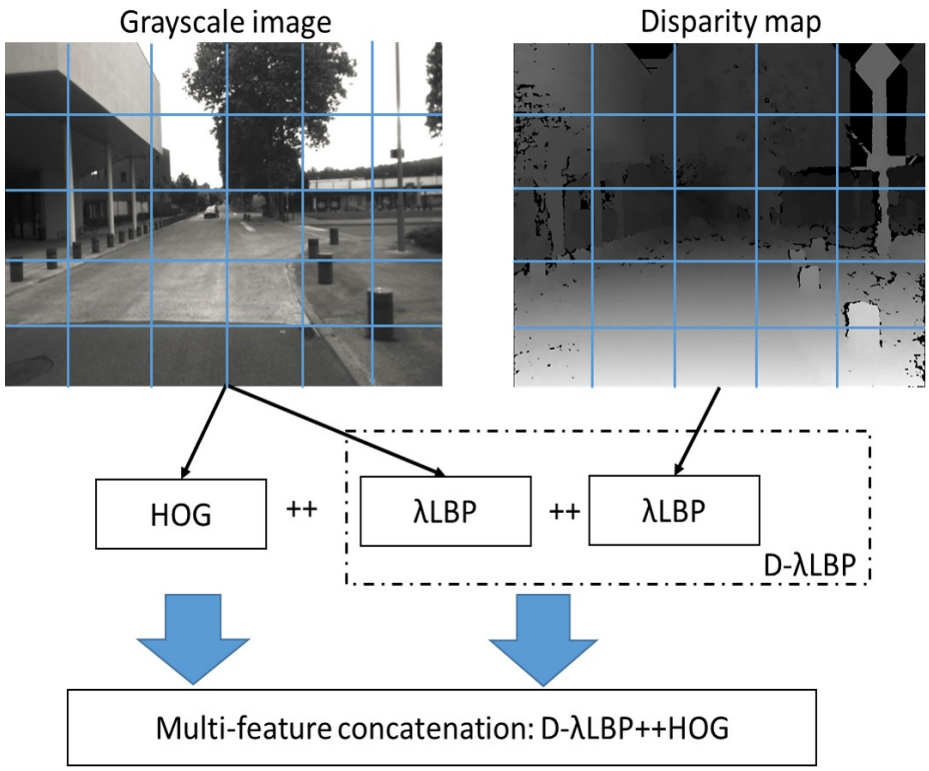


Figure 3-1: Multi-feature built from gray-scale image and disparity map. Features are firstly extracted from each image block and then concatenated together. The symbol "++" means concatenation.

then should improve place recognition performance. In this chapter, stereo images are used for visual place recognition. A novel localization approach is then proposed which uses multi-feature fusion by combining HOG and binary features (λLBP), as shown in Fig.3-1. HOG features are obtained from gray-scale image while λLBP features are built from both gray-scale image and disparity map. Noted that the features are first extracted from the blocks composing the gray-scle image and the disparity map, and then concatenated. We extend the application of λLBP descriptor to disparity map in order to incorporate disparity information in image representation by simply concatenating the two descriptors (λLBP from gray-scale image and λLBP from disparity map). This produces a new descriptor is named D-λLBP. The integration of disparity information in image representation provides depth information which should be helpful for place recognition, especially in complex environment situation. Indeed, image description using features λLBP and HOG and the depth information will permit to reduce perceptual aliasing problems related to

visual place recognition. As it will be shown in our experiments, features combination permits to achieve better recognition performance than single feature. Also the performance of place recognition is compared with the state-of-the-art FAB-MAP algorithm: the achieved F_1 scores on four tested datasets using our approach are better than those resulted from FAB-MAP method. Furthermore, considering high dimensional multi-features comparison is time-consuming, locality sensitive hashing is applied on multi-features to speed up the process of features comparison and image matching.

The most important contributions introduced in this chapter are the following:

- An innovative method for place recognition based visual localization using multi-feature descriptor (D-λLBP++HOG) extracted from gray-scale image and disparity map. The proposed multi-feature descriptor takes advantage of texture, depth and shape information and hence performs better than single feature (see Section 3.5.2).

- The impact image block size for the binary descriptors is studied. Binary descriptor extracted from small block has better discriminative ability in local details of different locations, while considering large block size for image representation may cause loss of some discriminative information (see Section 3.5.1).

- A speeding-up of the place recognition method is achieved by approximating the euclidean distance between features with hamming distance over bit-vectors obtained by Locality Sensitive Hashing (see Section 3.5.3).

The rest of this chapter is organized as follows. Firstly, the LBP descriptor and several of its variants as well as HOG feature are introduced in Section 3.2. Then, in Section 3.3, the proposed approach is described in detail. Section 3.4 deals with the presentation of the tested database and the used performance evaluation parameters. The obtained results are presented and discussed in Section 3.5. Finally, conclusions and future works close this chapter (Section 3.6).

3.2 Overview of used Image Descriptors

In this part, some of the state-of-the-art image descriptors used and compared in the proposed approach are described.

3.2.1 LBP (Local Binary Pattern)

LBP is a texture descriptor that codifies local primitives (such as curved edges, spots, flat areas) into a feature histogram. The original LBP operator labels the pixels of an image with decimal numbers, called Local Binary Patterns or LBP codes, which encode the local structure around each pixel [56].

As illustrated in Fig.3-2, each pixel gray-level value is compared with its eight neighbors in a 3×3 region by subtracting the center pixel value. The resulting strictly negative values are encoded with 0 and the others with 1. A binary number is obtained by concatenating all these binary codes, and its corresponding decimal value is used for labeling the central pixel. In Fig.3-3, examples of neighborhood used for LBP operator are illus-

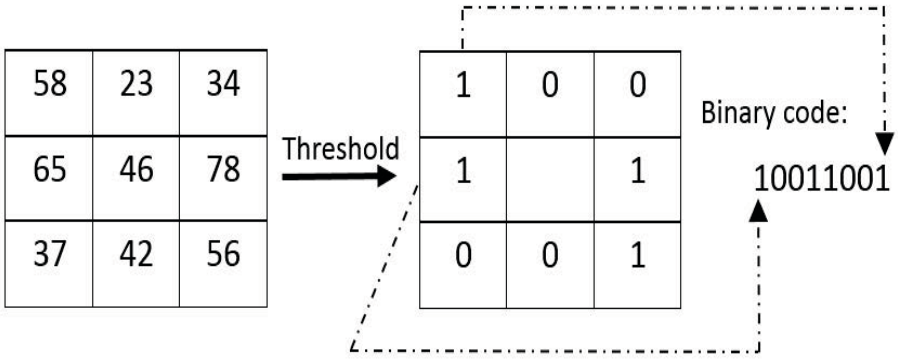


Figure 3-2: Illustration of the basic LBP operator

trated. The generalized LBP definition uses P sample points evenly distributed on a radius R around a center pixel located at (x_c,y_c). The position (x_p, y_p), of the neighboring points, where $p \in \{0,...,P-1\}$ is given by:

$$(x_p, y_p) = (x_c + Rcos(2\pi p/P), y_c - Rsin(2\pi p/P)) \tag{3.1}$$

The local binary code for the position (x_c, y_c) can be computed by comparing the gray-scale

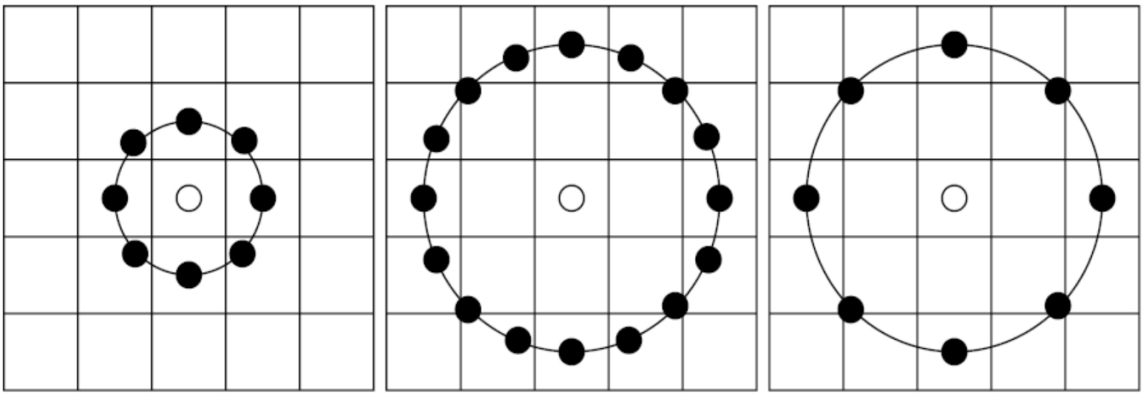


Figure 3-3: Examples of (P,R) neighborhood used to compute LBP: (8,1), (16,2) and (8,2)
.

value g_c of this center pixel located at (x_c, y_c) and the gray-scale values g_p of its neighbor pixels located at (x_p, y_p) where $p \in \{0, ..., P-1\}$. The value of the LBP code of the center pixel at position (x_c, y_c) is given by:

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p \tag{3.2}$$

where s is the Heaviside function:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & otherwize \end{cases} \tag{3.3}$$

The operator $LBP_{P,R}$ produces 2^P different output values, corresponding to 2^P different binary patterns formed by the P pixels in the neighborhood. Although this method can capture the relations of nearby and adjacent pixels, it leads to a large data dimension.

Ojala et al. [79] further propose an "uniform patterns" to reduce the dimension of LBP feature while keeping its discrimination power. For this, an uniformity measure of a pattern is used: U ("pattern") is the number of bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. The U value of an LBP pattern can be computed by:

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)| \tag{3.4}$$

Uniform LBP patterns refer to the patterns which have limited transitions or discontinuities ($U \leq 2$) in the circular binary representation. For instance, 11111111 (0 transitions) and 01110000 (2 transitions) are both uniform whereas 11001001 (4 transitions) and 01010010 (6 transitions) are not. Thus, for P neighborhood pixels, a uniform $LBP_{P,R}$ operator produces $P(P-1)+3$ possible distinct uniform LBP patterns. After the uniform LBP patterns are identified, for an image with size $N \times M$, a histogram is built which can be used as the image feature to represent the image texture :

$$h(l) = \sum_{i=1}^{N} \sum_{j=1}^{M} f\left(LBP_{P,R}(i,j), l\right), \; l \in [0, L], \tag{3.5}$$

$$f(x,y) = \begin{cases} 1, & x = y \\ 0, & otherwize \end{cases} \tag{3.6}$$

where L is the maximal LBP pattern value. The length of the histogram is a $P(P-1)+3$.

3.2.2 CLBP (Complete Local Binary Pattern)

LBP feature considers only signs of local differences (i.e. difference of each pixel with its neighbors) whereas CLBP feature [42] considers both magnitude (M) and sign (S) of local differences as well as original center gray level value (C) . Consequently, three operators, namely CLBP_M, CLBP_S and CLBP_C, are used to code the magnitude, sign and center gray level.

Given the gray-scale value g_c of the center pixel (x_c, y_c) and its P circularly and evenly spaced neighbors with gray-scale value g_p, $p \in \{0, ..., P-1\}$, the difference between g_c and g_p can be simply calculated using $d_p = g_p - g_c$. The local difference vector [d_0, d_1, $\cdots$, d_{P-1}] characterizes the image local structure at (x_c, y_c). Because the central gray level g_c is removed in local difference vector, [d_0, d_1, $\cdots$, d_{P-1}] is robust to illumination changes and is more efficient in pattern matching. d_p can be further decomposed into two components:

$$d_p = s_p * m_p \tag{3.7}$$

$$m_p = |d_p|, \quad s_p = \begin{cases} 1, & d_p \geq 0 \\ -1, & d_p < 0 \end{cases} \tag{3.8}$$

where s_p is the sign component of d_p and m_p is the magnitude component of d_p.

CLBP_M is used to code the magnitude information of local differences:

$$CLBP\_M_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} t(m_p, c)2^p, \quad t(x,T) = \begin{cases} 1, & x \geq T \\ 0, & x < T \end{cases} \tag{3.9}$$

where T is a threshold which is set to the mean value of the m_p values from the whole image.

CLBP_S is the same as the original LBP and is used to code the sign information of local differences:

$$CLBP\_S_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} t(s_p, 0)2^p, \quad t(x,T) = \begin{cases} 1, & x \geq T \\ 0, & x < T \end{cases} \tag{3.10}$$

CLBP_C is used to code the information of original center gray level value:

$$CLBP\_C_{P,R}(x_c, y_c) = t(g_c, c_I), \quad t(x,T) = \begin{cases} 1, & x \geq T \\ 0, & x < T \end{cases} \tag{3.11}$$

where the threshold c_I is set to the average gray level of the input image.

The dimension of the histograms corresponding to CLBP_S and CLBP_M is 2^P, while the dimension of CLBP_C is 2. The CLBP_C only uses the center gray level value which can be easily affected by the changing of viewpoints or illumination. Therefore, in our work, only the histograms of CLBP_S and CLBP_M codes are computed and then concatenated together to construct CLBP feature. Thus, the final dimension of CLBP feature is 2^{P+1}.

3.2.3 CSLBP (Center-symmetric local binary patterns)

CSLBP [43] is another modified version of LBP. CSLBP produces shorter feature set than LBP, but it is also a first order local pattern in center symmetric direction and it ignores the central pixel information. CSLBP is closely related to the gradient operator, because it compares the gray levels of pairs of pixels in centered symmetric directions instead of comparing the central pixel to its neighbors. In this way, CSLBP feature takes advantage of the properties of both LBP and gradient based features.

For an even number P of neighboring pixels distributed on radius R, CSLBP operator produces $2^{P/2}$ possible distinct patterns. The operator is given by:

$$CSLBP_{P,R}(x_c, y_c) = \sum_{i=0}^{(P/2)-1} s(|g_i - g_{i+(P/2)}|)2^i \tag{3.12}$$

$$s(x) \begin{cases} 1, & x \geq T \\ 0, & otherwise \end{cases} \tag{3.13}$$

where g_i and $g_{i+(P/2)}$ are the gray values of center-symmetric pairs of pixels. T is used to threshold the gray-level difference so as to increase the robustness of CSLBP feature on flat image regions. Since the gray levels are normalized in [0,1], the authors of paper [43] recommend to use small value for T.

It should be noticed that CSLBP is closely related to gradient operator, because like some gradient operators it considers gray level difference between opposite pixels in a neighborhood.

Given an image of size $N \times M$, after the computation of CSLBP patterns, a histogram is built to represent the texture image:

$$h(l) = \sum_{i=1}^{N} \sum_{j=1}^{M} f\left(CSLBP_{P,R}(i,j), l\right), \;\; l = 0,1,2,3,\cdots,2^{P/2}-1 \tag{3.14}$$

$$f(x,y) = \begin{cases} 1, & x = y \\ 0, & otherwize \end{cases} \tag{3.15}$$

By construction, the length of the histogram resulting from CSLBP feature is $2^{P/2}$.

3.2.4 CSLDP (Center-Symmetric Local Derivative Pattern)

CSLDP operator [112] is a second order derivative pattern in center symmetric direction. CSLDP captures more information by encoding the relationship between central pixel and center symmetric neighbors. Moreover, CSLDP has shorter length than LBP.

For an even number P of neighboring pixels distributed on radius R, CSLDP operator produces $2^{P/2}$ possible distinct patterns and is defined as:

$$CSLDP_{P,R}(x_c, y_c) = \sum_{i=0}^{P/2-1} t[(g_i - g_c), (g_c - g_{(i+(P/2))})]2^i \tag{3.16}$$

where g_i, $g_{(i+(P/2)}$ are gray-scale values of neighborhood pixels in center symmetric direction. g_c corresponds to the gray value of central pixel located at (x_c, y_c). The threshold function $t(\cdot,\cdot)$ is used to determine the type of local pattern transition and is defined as:

$$t(x_1, x_2) = \begin{cases} 1, & x_1 \cdot x_2 \leq 0 \\ 0, & x_1 \cdot x_2 > 0 \end{cases} \tag{3.17}$$

A CSLDP pattern encodes the second order center symmetric derivatives at pixel (x_c, y_c) along 0°, 45°, 90° and 135° directions. They can be represented as:

$$\begin{cases} CSLDP_{0^\circ}(x_c, y_c) = t[(g_0 - g_c), (g_c - g_4)] \\ CSLDP_{45^\circ}(x_c, y_c) = t[(g_1 - g_c), (g_c - g_5)] \\ CSLDP_{90^\circ}(x_c, y_c) = t[(g_2 - g_c), (g_c - g_6)] \\ CSLDP_{135^\circ}(x_c, y_c) = t[(g_3 - g_c), (g_c - g_7)] \end{cases} \tag{3.18}$$

The CSLDP histogram construction method is the same as for CSLBP, and its histogram length is also $2^{P/2}$.

3.2.5 XCSLBP (extended CSLBP)

The work in [95] proposes a new LBP variant called XCSLBP (eXtended CSLBP), which compares the gray values of pairs of center symmetric pixels considering the central pixel, without increasing histogram length. This combination makes the resulting descriptor robust to illumination changes and noise. For an even number P of neighboring pixels distributed on radius R, XCSLBP is expressed as:

$$XCSLBP_{P,R}(x_c, y_c) = \sum_{i=0}^{(P/2)-1} s(g_c^2 + g_{i+(P/2)}(g_i - 2g_c))2^i, \tag{3.19}$$

where the threshold function s, which is used to determine the types of local pattern transition, is defined as:

$$s(x) = \begin{cases} 1, & (x \geq 0) \\ 0, & otherwise \end{cases} \tag{3.20}$$

where g_i and $g_{i+(P/2)}$ are the gray values of center symmetric pixels. XCSLBP operator produces histograms with a length of $2^{P/2}$.

3.2.6 HOG (Histograms of Oriented Gradients)

Besides LBP and its variants, another histogram feature named HOG has also been widely accepted as one of the best features to capture the edge or local shape information. HOG feature is proposed by Dalal et al. [25] and widely used to detect objects in computer vision. The essential idea of HOG feature is that the shape or appearance of local object can be described by the distribution of intensity gradients and edge directions [88]. HOG descriptor is a one-dimensional histogram of gradient orientations of intensity in local regions that can represent object shape.

As shown in Fig.3-4, HOG divides the image into small connected blocks, and for each block, a histogram of gradient directions for the pixels within the block is computed. The combination of these cell histograms represents the feature vector. At each pixel, the gradient is a 2D vector with a real-valued magnitude and a discretized direction (9 possible directions uniformly distributed in [0, π]). During the construction of the integral image

of HOG, the feature value at each pixel is treated as a 9D vector, and the value at each dimension is the interpolated magnitude value at the corresponding direction. Since HOG takes adjacent pixel gradients information as basis to extract features, it is robust to changes in geometry and is not easily affected by local lighting conditions.

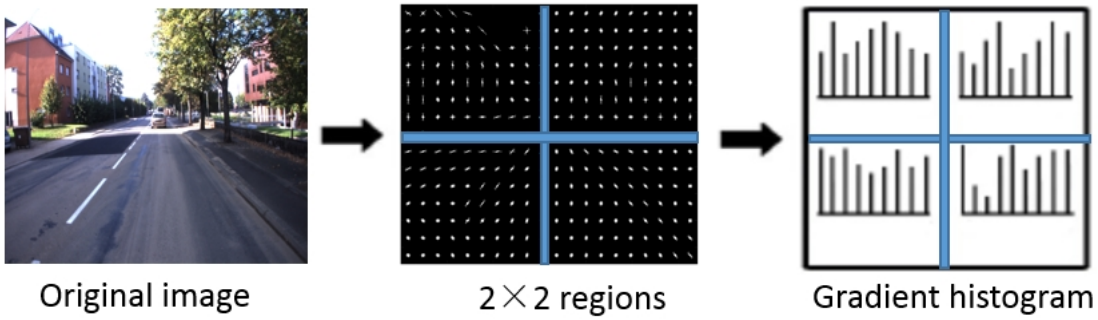


Figure 3-4: Example of HOG feature.

3.3 Overview of Proposed Approach

In this section, a robust visual localization based on multi-feature combination is developed. The general principle is to find the image that best matches the current acquired one, among a set of previously acquired and GPS-tagged training images.

The whole system includes an off-line phase and an on-line phase. In the off-line phase, a set of GPS tagged training image pairs (left and right images) $I^{train} = \{I_j^{train}\}_{j=1}^{N^{train}}$ are firstly acquired, where N^{train} is the number of training image pairs. After image preprocessing (see Section 3.3.1), multi-feature set $V^{train} = \{v_j^{train}\}_{j=1}^{N^{train}}$ is extracted from the training database (see Sections 3.3.2 and 3.3.3) , where v_j^{train} is the multi-feature extracted from the training image pair I_j^{train}. In on-line phase, multi-feature v_i^{test} is extracted from current image pair I_i^{test}, and then compared with each multi-feature of V^{train} based on euclidean distance. The computed distances are then used to select the best candidate (see Section 3.3.4), smaller the distance is, higher similarity between the images will be. A distance ratio SS between the two best candidates (i.e. corresponding to the two minimum computed distances) is considered for matching validation (see Section 3.3.5). If the ratio SS is lower than or equal to a threshold Th, the first best image candidate (with the lower

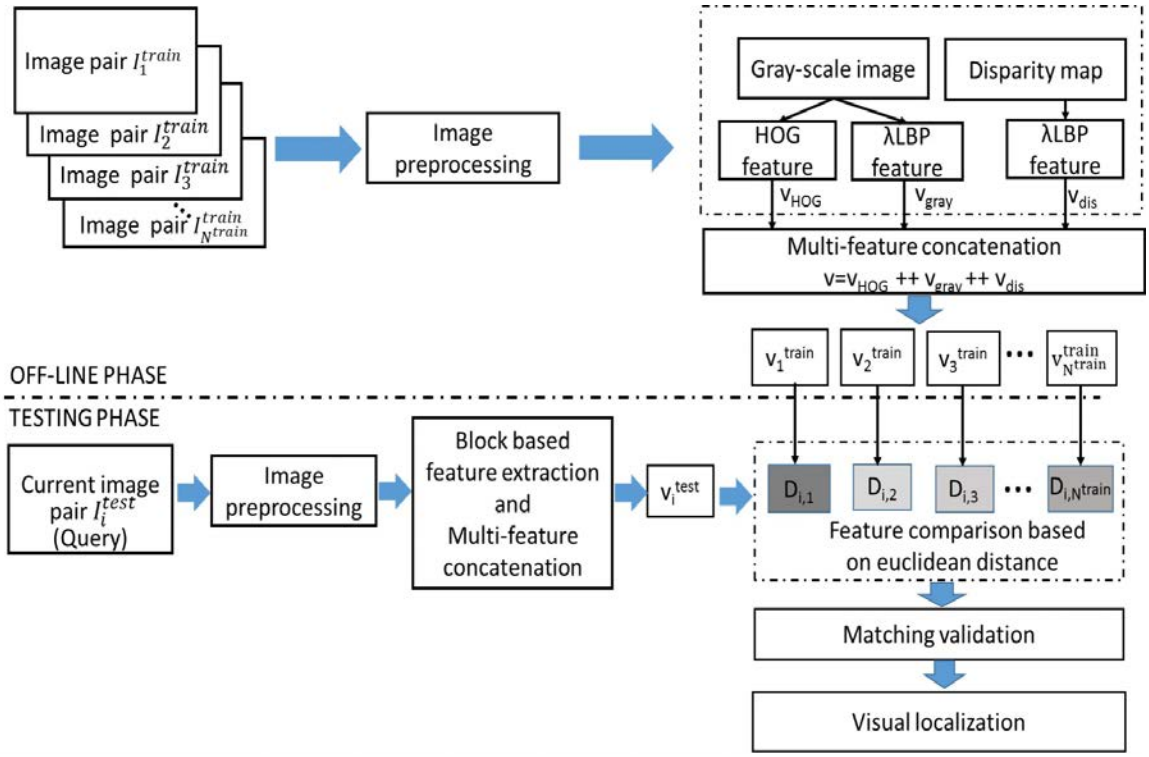


Figure 3-5: The process of the proposed place recognition based visual localization.

matching distance) is confirmed as positive, otherwise it is regarded as negative (in this case, no matching result is conserved). When a matching is confirmed as positive, the current position can be obtained from the matched GPS-tagged training image (see Section 3.3.6).

As illustrated in Fig.3-5, the overall approach comprises six stages:

1. **Image preprocessing**: This step consist of down-sampling and contrast-limited adaptive histogram equalization (detailed in Section 3.3.1).

2. **Block based feature extraction**: λLBP feature is extracted from grayscale image and disparity map; HOG feature is extracted from grayscale image (detailed in Section 3.3.2).

3. **Multi-feature concatenation**: The final multi-feature D-λLBP++HOG is obtained by concatenating λLBP and HOG feature. (detailed in Section 3.3.3)

4. **Feature comparison and image matching**: Based on the extracted multi-feature

descriptors, image matching is conducted through multi-feature comparison using euclidean distance (detailed in Section 3.3.4).

5. **Final Matching validation**: According to the distance ratio of the top two best candidates, image matching result is validated (detailed in Section 3.3.5).

6. **Visual localization**: The vehicle current position can be obtained through the matched GPS-tagged training image (detailed in Section 3.3.6).

3.3.1 Image preprocessing

Image preprocessing is composed of two parts: down-sampling and contrast-limited adaptive histogram equalization (CLAHE).

Down-sampling permits to reduce the original image size, which makes feature extraction faster. In fact, it has been already proved in [89] that high resolution images are not more helpful than lower resolution ones. Therefore, down-sampling is the first step before feature extraction. As it is well known, illumination has significant influence on

Figure 3-6: Image preprocessing using contrast-limited adaptive histogram equalization(CLAHE). The left image is processed using CLAHE and the prepossessing result is the right image.

outdoor image appearance. Therefore, another applied image preprocessing is contrast-limited adaptive histogram equalization (CLAHE), which permits to enhance the contrast of the gray-scale image by transforming the values using contrast-limited adaptive histogram equalization [83]. Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to gain higher contrast. This contrast, especially in homogeneous areas, can be limited to avoid amplifying any noise that might be present in the image. On the same time, it also decreases the shadow

influence. An image example after applying contrast-limited adaptive histogram equalization can be seen in Fig.3-6. It is obvious that CLAHE prepossessing improves the image contrast and makes the image more brighten (especially in some dark parts).

3.3.2 Block based feature extraction

Concept of block based approach

Traditionally, local descriptors are calculated on full images, which can keep the size of the feature database reasonably low. However, local image areas of interest would be ignored as the full image feature extraction does not contain enough local discriminative information.

With respect to local properties and enhanced image representation ability, image features are extracted from small image blocks (sub-image areas) without any segmentation and then these independent feature descriptors are concatenated to obtain final image feature. To illustrate the block based feature extraction process, it is applied on an example in Fig.3-7. Block based approach (that relies on image blocks) can address spatial properties of images. It can be used for any histogram descriptors.

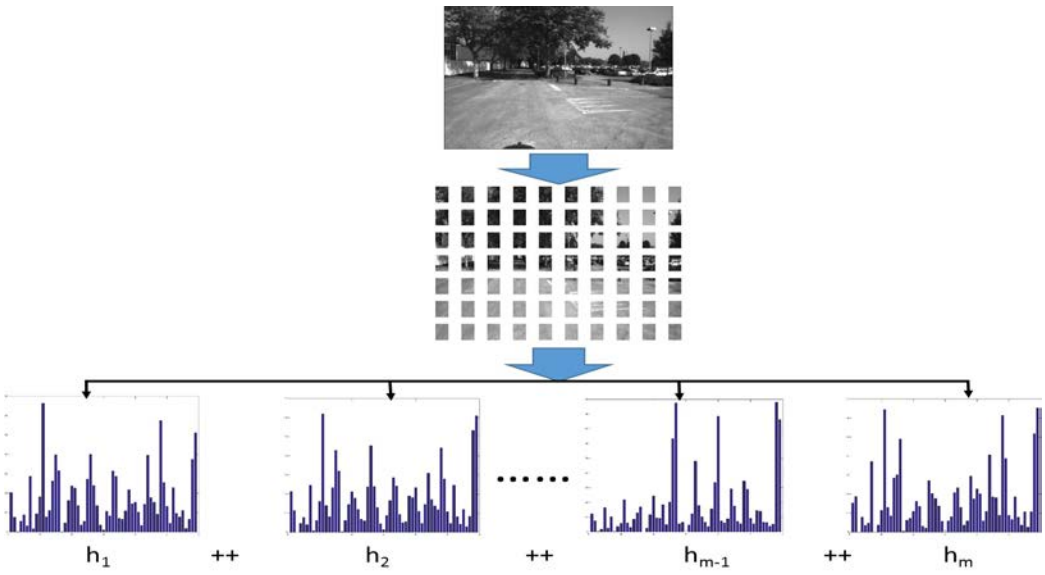


Figure 3-7: An example of block based local binary descriptor extraction. Features are extracted from each image block firstly and then concatenated together. Here, image blocks are non-overlapped and do not need any image segmentation.

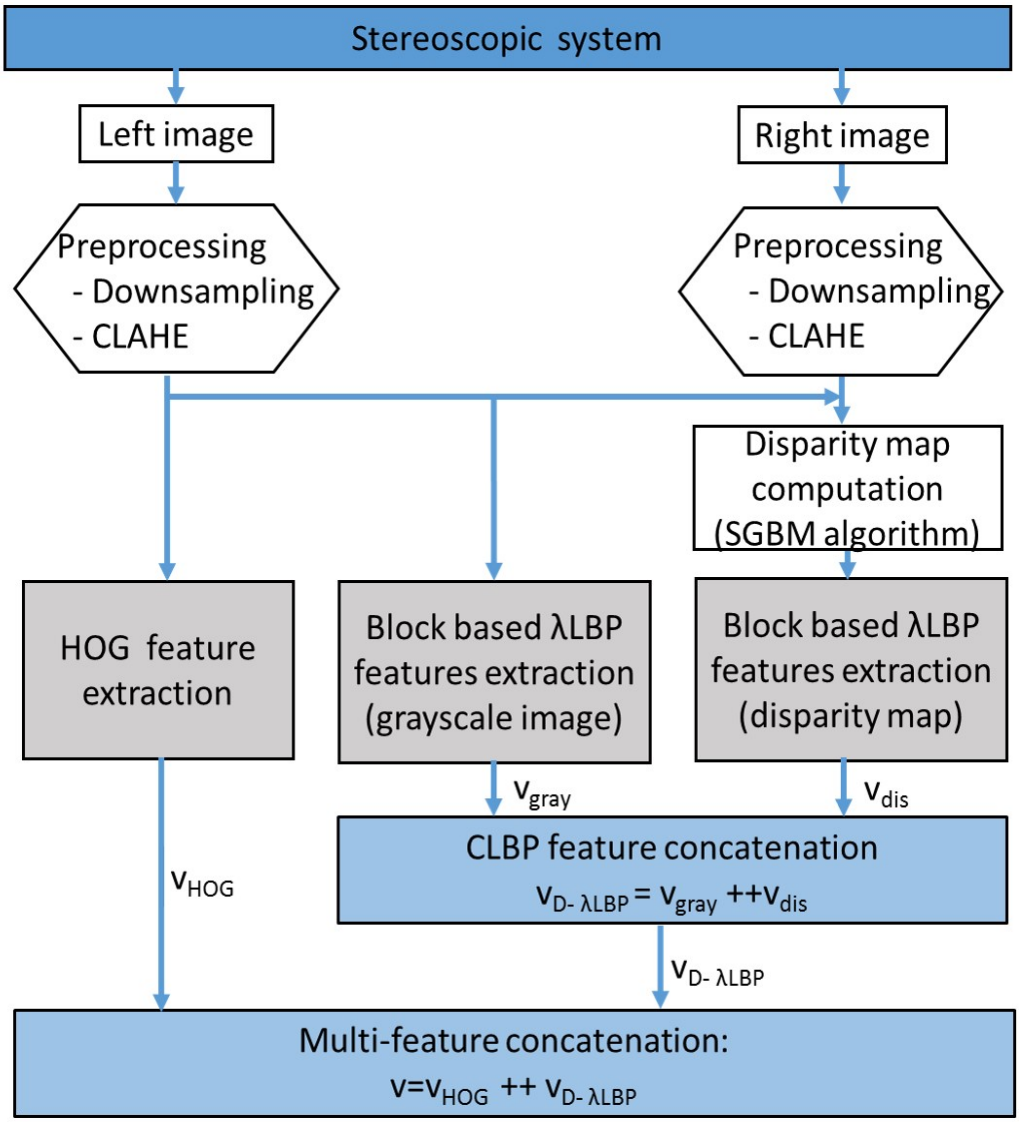


Figure 3-8: Block based feature extraction procedure (applied to each images pair of the training database for the off-line phase and to current images pair for the testing phase)

Block based feature extraction

After image preprocessing, features are extracted, as illustrated in Fig.3-8. λLBP feature is extracted from gray-scale image and disparity map independently. While HOG feature v_{HOG} is extracted from gray-scale image. For both λLBP or HOG, the features are extracted based on image blocks. In order to facilitate the process of block based feature extraction, image blocks in the full image have the same size. The influence of different block sizes will be studied in the Section 3.5.1. Image parts that cannot satisfy a whole

block will be ignored.

- **λLBP feature extraction**

  λLBP feature from gray-scale image and disparity map are obtained using the following equations:

$$\begin{cases} v_{gray} = h_1^{gray} + + h_2^{gray} + + ... + + h_m^{gray} \\ v_{dis} = h_1^{dis} + + h_2^{dis} + + ... + + h_n^{dis} \end{cases} \tag{3.21}$$

  where v_{gray} is a vector which stores the λLBP feature obtained from gray-scale image. v_{dis} stores the λLBP obtained from disparity map. m and n are the image block numbers of gray-scale image and disparity map respectively. $h_i^{gray}(i \in [1,2,\cdots,m])$ is the λLBP histogram of the i^{th} block of the grayscale image and $h_i^{dis}(i \in [1,2,\cdots,n])$ is the λLBP histogram of the i^{th} block of the disparity map. In our work, the disparity map is calculated using the SGBM (Semi-Global Block Matching) algorithm [44]. Using this SGBM method, there are some useless parts (" black areas"), for which no depth information is computed, especially on the left and right sides of the disparity map. In these " black areas", λLBP operator is not applied, therefore these useless parts are simply removed. Thus, due to the removing of the " black areas" in the disparity map, m and n are not identical.

  By using the block based approach, the features v_{gray} and v_{dis} are extracted from gray-scale image and disparity map respectively. Then, D-λLBP feature can be computed by concatenating v_{gray} and v_{dis}:

$$v_{D-\lambda LBP} = v_{gray} + + v_{dis} \tag{3.22}$$

- **HOG feature extraction**

  HOG feature is also computed for each image block of the gray-scale image. The obtained HOG features from all the image blocks are then concatenated:

$$v_{HOG} = h_1^{hog} + + h_2^{hog} + + ... + + h_m^{hog} \tag{3.23}$$

Here, $h_i^{hog}(i \in [1,2,\cdots,m])$ is the HOG feature extracted from the i^{th} image block. It should be noted that HOG feature adopts the same image block size as the λLBP feature extraction from gray-scale image, therefore the number of image blocks is same.

3.3.3 Multi-feature concatenation

In order to take advantage of the different features, D-λLBP and HOG are combined together to represent the image. Since the D-λLBP and HOG are two independent features, we simply consider that they have the same weight in the role of place recognition. The final multi-feature can be obtained easily through concatenation using the following equation:

$$v = v_{D-\lambda LBP} + + v_{HOG} \tag{3.24}$$

Using this method, a multi-feature set $V^{train} = \{v_j^{train}\}_{j=1}^{N^{train}}$ of all training image pairs $\{I_j^{train}\}_{j=1}^{N^{train}}$ is obtained. For a current testing image pair I_i^{test}, a multi-feature v_i^{test} is also obtained. Then the image matching is conducted based on the euclidean distance comparison between the multi-feature v_i^{test} of the current testing image and all training image multi-features $v_j^{train}(j = [1,2,\cdots,N^{train}])$ from the training images dataset.

3.3.4 Feature comparison and image matching

Feature comparison is performed based on the euclidean distance between features. Each testing image multi-feature v_i^{test} is compared with all the training images multi-features $v_j^{train}(j = [1,2,\cdots,N^{train}])$ of the training database.

The distance $D_{i,j}$ between the multi-feature $v_i^{test}(i = [1,2,\cdots,N^{test}])$ of the testing image and multi-feature vector $v_j^{train}(j = [1,2,\cdots,N^{train}])$ of a training image is computed as follows:

$$D_{i,j} = \|v_i^{test} - v_j^{train}\|^2 \tag{3.25}$$

where $\|\cdot\|$ denotes the euclidean norm.

In fact, small distance means high similarity. Based on euclidean distance, image

matching candidates are searched. After distance computation, for the testing image, the two minimum distances ($D_{i,m1}$ and $D_{i,m2}$) and their corresponding training images (the two best candidates) are conserved.

3.3.5 Final matching validation

For a given current image pair I_i^{test}, the validation of matching candidate from the training database is based on the ratio SS_i, calculated as follows:

$$SS_i = \frac{D_{i,m1}}{D_{i,m2}} \tag{3.26}$$

where $D_{i,m1}$ and $D_{i,m2}$ are respectively the first and second minimum distances between the current image multi-feature v_i^{test} and the multi-features $\{v_j^{train}\}_{j=1}^{N^{train}}$ of all the training images:

$$\begin{cases} D_{i,m1} = \min\limits_{j}\{D_{i,j}\} \\ D_{i,m2} = \min\limits_{j(j\neq m1)}\{D_{i,j}\} \end{cases} \tag{3.27}$$

As said before, lower the distance is, more similar the images are. The potential matching candidate is the image m_i (the one giving the lower distance with the testing image). However, if the second best matching candidate provides a distance very close to the first one, this means that the matching algorithm provides two confused solutions. In this case, we propose to ignore the matching result and consider that the testing image has no matching image. For that, a threshold Th is applied to the ratio SS_i, which takes its values in the range [0 1].

The last decision is as follows: if SS_i is lower than or equal to the threshold Th, then the pair $(i,m1)$ is considered as positive, and the pair is matched. Otherwise, the pair is considered as negative and the pair is ignored.

3.3.6 Visual localization

After image matching result is successfully validated, the vehicle can localize itself through the matched training image position. Since the training images are tagged with the GPS or

pose information, the vehicle can get its position information by assimilating its position to the GPS position of the training image matched with the current testing image. This is a topological level localization, that is, the system simply identifies the most likely location. Therefore, this is not a very accurate metric localization, because the training and testing trajectories are not exactly same.

It should be noted that, some places can not be localized at the situation of validation failure (negative matching case).

3.3.7 Algorithm of multi-feature based visual localization

The approach for place recognition based visual localization proposed in this chapter is summed up in Algorithm 3.1. The multi-features are constructed firstly from training database, this is off-line processing. Then multi-feature built from the current image pair is compared with the whole multi-features from training database to obtain the euclidean distance vector. Based on the this, image matching result is validated for the final visual localization.

3.4 Experimental Setup

3.4.1 Datasets and ground-truth

The proposed method is tested on four different datasets (UTBM-1, UTBM-2, KITTI 05 and KITTI 06).

The taken route for UTBM-1 dataset is shown in Fig.3-9 (a): the experimental vehicle traversed about 4 km in a typical outdoor environment. Three typical areas were traversed: urban city road (area A), lots of factories building (area B) and a nature scene surrounding a lake (area C). The training and testing data were collected at different times respectively in 2014/9/11 and 2014/9/5. The training database is composed of 849 images while the testing database is composed of 819 images. The average distance between two successive frames was around 3.5 m. To tag the training images, GPS position of each image is obtained by a RTK-GPS receiver.

Algorithm 3.1 Place recognition based visual localization using multi-feature.

Inputs:

$\{I_j^{train}\}_{j=1}^{N^{train}}$ {training image pairs database (left and right images)};
$\{I_i^{test}\}_{i=1}^{N^{test}}$ {testing images pairs (left and right images)};
N^{train}, N^{test} {training and testing images number};

Outputs:

SS{distance ratio}; Vehicle position

Proposed Algorithm:

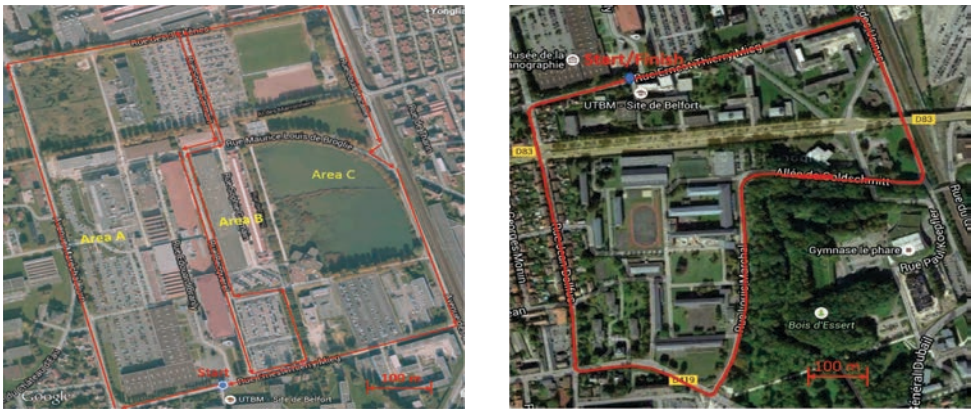
/* OFF-LINE PHASE */
/* Training images multi-feature extraction and concatenation; {Section 3.3.2 and 3.3.3} */
for $j \leftarrow 1$ to N^{train} **do**
  $v_{gray,j}^{train}, v_{HOG,j}^{train} \leftarrow$ Block based λLBP and HOG features extraction from gray-scale images;
  $v_{dis,j}^{train} \leftarrow$ Block based λLBP feature extraction from disparity map;
  $v_j^{train} \leftarrow v_{gray,j}^{train} ++ v_{dis,j}^{train} ++ v_{HOG,j}^{train}$; //Multi-feature concatenation.
end for

/* ON-LINE PHASE */
/* Feature comparison */
for $i \leftarrow 1$ to N^{test} **do**
  $v_i^{test} \leftarrow v_{gray,i}^{test} ++ v_{dis,i}^{test} ++ v_{HOG,i}^{test}$; //Multi-feature computation for the current testing image pair.
  for $j \leftarrow 1$ to N^{train} **do**
    $D_{i,j} = \|v_i^{test} - v_j^{train}\|^2$; Euclidean distance computation between the multi-feature v_i^{test} of the testing image and the multi-feature v_j^{train} in the training database.
  end for

  /* Matching validation and localization */
  $SS_i = \frac{\min_j\{D_{i,j}\}}{\min_{j(j\neq m1)} D_{i,j}}$; $m1$ is the index of the first best candidate.
  if $SS_i <= Th$
    Matching validation is positive;
    Vehicle position $\leftarrow$ the matched training image position
  if $SS_i > Th$
    Matching validation is negative;
    Vehicle position $\leftarrow$ NaN (no position result)
end for

The UTBM-2 dataset (Fig.3-9 (b)) consists of a 2.3 km route in Belfort city downtown acquired in 2014/9/5. The first traversal to acquire training dataset was performed in the

morning and the second one was conducted in the afternoon to acquire testing dataset. Each travel time across this dataset was approximately 20 minutes. The training database is composed of 540 images while the testing database is composed of 520 images. The GPS information of each image is also collected. The popular KITTI benchmark dataset is also



(a) Trajectory of UTBM-1 dataset (b) Trajectory of UTBM-2 dataset

Figure 3-9: Vehicle paths for the UTBM-1 and UTBM-2 datasets. Source: Google Maps

used to test our proposal. The KITTI Odometry dataset has 22 sequences containing a total of 44182 stereo images (39.2 km). These sequences include environments with different characteristics and challenging situations such as perceptual aliasing, changes on scene, etc. Among them, the datasets KITTI 05 and KITTI 06 that contain loop closures were selected to evaluate our method. There are 2761 and 1101 images in KITTI 05 and KITTI 06 datasets respectively.

For UTBM-1 and UTBM-2 datasets, ground-truth was constructed by manually finding pairs of frame correspondences according to the GPS data. While the KITTI dataset ground-truth was built according to the pose information [6].

3.4.2 Image preprocessing and feature extraction

In our work, for faster feature extraction, the original color images were down-sampled into half scale size grayscale image. That means images in dataset UTBM-1 and UTBM-2 were resized to 640 $\times$ 480 and the images in dataset KITTI 05 and KITTI 06 were resized to 613$\times$235.

In order to reduce the illumination influence on the outdoor image appearance, contrast-limited adaptive histogram equalization (CLAHE) method was used (see Section 3.3.1).

Moreover, as a pair of images is acquired at each instant, a disparity map can be computed easily using the SGBM (Semi-Global Block Matching) algorithm [44].

After image preprocessing, binary descriptors (LBP, CLBP, CSLBP, CSLDP and XCSLBP) are extracted with the following parameters: 8 sampling points and 3 pixels radius. HOG descriptor is extracted from the grayscale images. To capture large-scale spatial information, the cell size of HOG is 32×32. The number of cells in each block is specified as a 2-element vector.

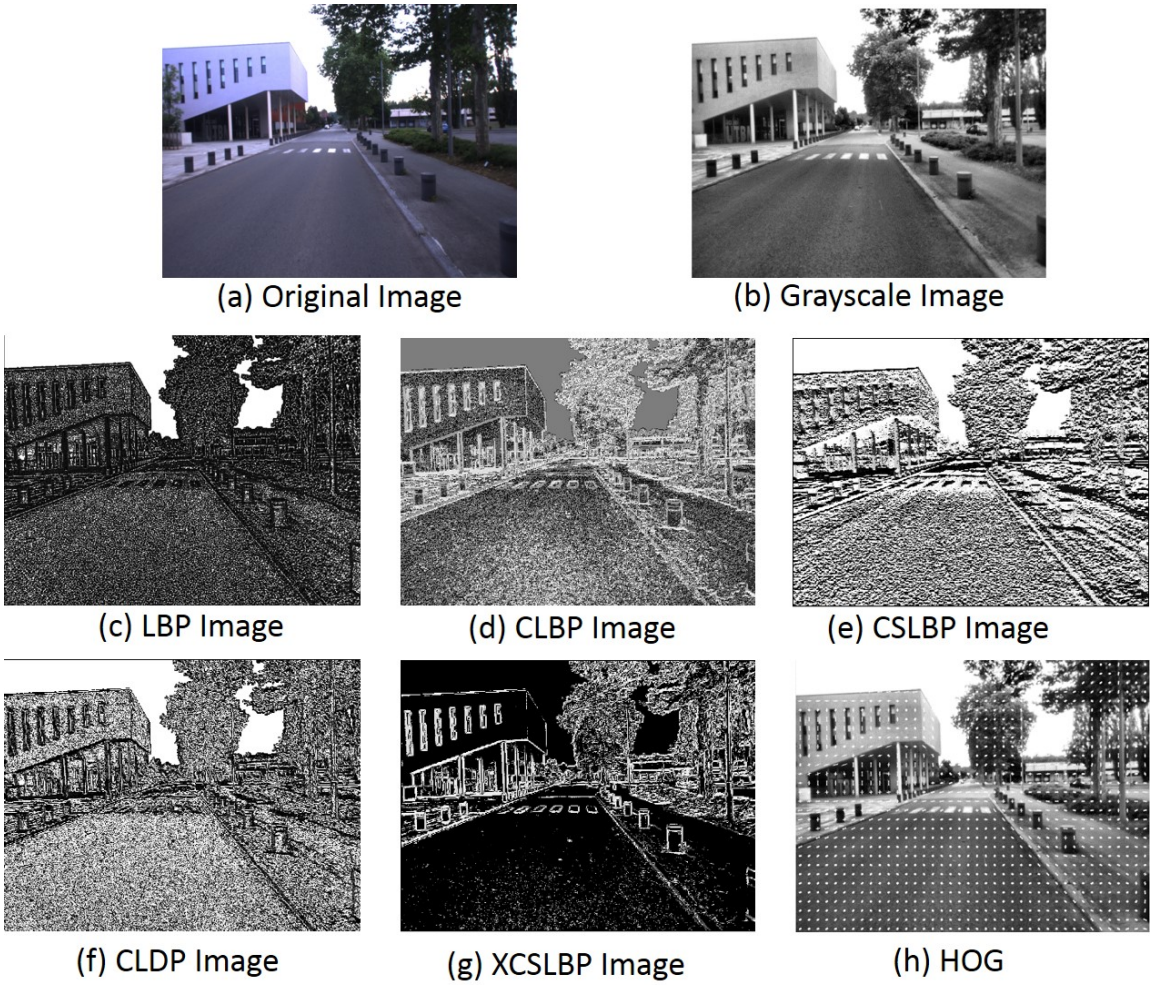


Figure 3-10: Example of gray-scale image and its corresponding local binary images (LBP, CLBP, CSLBP, CSLDP and XCSLBP) and HOG feature.

An example of extracted image features can be seen in Fig.3-10. It can be seen that

the local binary features pay more attention to texture information. It can also be noted that CSLBP and XCSLBP perform better than LBP. HOG feature depicts object shape information in the image. Therefore, combining LBP and HOG features could bring more information and make place(scene) better described.

3.4.3 Performance evaluation

Precision-recall characteristics and F_1 score are widely used to show the effectiveness of image retrieval method. Therefore, our evaluation methodology is based on precision-recall curves and F_1 score. In our experiments, the training image number is larger than or equal to the testing image number, thus each testing image has a ground-truth matching. Therefore, among the positives, there are true positives (correct results among successfully validated images matching candidates) and false positives (wrong results among successfully validated images matching candidates). The sum of the true positives and false positives is the total retrieved images number.

More specifically, precision is the ratio of true positives over the retrieved images number (number of all the successfully validated image matching candidates), and recall is the ratio of true positives over the total testing images:

$$Precision = \frac{\text{Number of true positives}}{\text{Number of retrieved images}} \times 100\%$$

$$Recall = \frac{\text{Number of true positives}}{\text{Number of total testing images}} \times 100\%$$

The final curve is computed by varying the threshold Th (applied to the ratio SS) in a linear distribution between 0 and 1, with the calculation of the corresponding values of precision and recall. 100 values of threshold Th are considered to obtain well-defined curves. When the threshold is set to 1, the candidates whose ratio is below or equal 1 are positives. In this case, the number of retrieved images is identical to the number of testing images. While when the threshold is 0, it means that the candidates whose ratio is below or equal 0 are regarded as positives. In this case, there is no retrieved image.

Precision relates the number of correct matches to the number of false matches, whereas

recall relates the number of correct matches to the number of missed matches. A perfect system would return a result where both precision and recall have a value of one. The F_1 score value is a single value that indicates the overall effectiveness of image retrieval method. Based on the precision and recall, F_1 score is defined as:

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.28}$$

3.5 Experiments and Results

Different aspects of our proposal are evaluated in the following sections. In Section 3.5.1, the performance of binary features (LBP and its variants) with and without disparity information is studied. In addition, the image block size influence for the binary feature D-CSLBP is investigated in Section 3.5.1. In Section 3.5.2, the effect of the multi-feature fusion proposed in our approach is analyzed. It is to note that the experiment results obtained in Sections 3.5.1 and 3.5.2, are based on euclidean distance. In Section 3.5.3, the efficiency of our LSH based visual recognition is checked: the execution time and recognition performance of our complete system are evaluated. Finally, visual localization at 100% recognition level is discussed in Section 3.5.4.

3.5.1 Comparison of the different binary features and image block sizes

Performance of different binary features

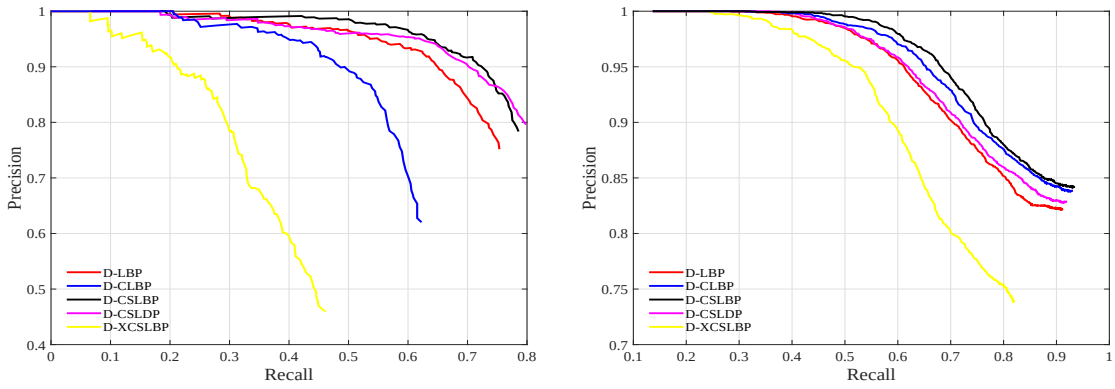
In this section, we compare binary features performance in two situations: with or without disparity map. Here the features are compared based on the euclidean distance.

Table 3.1 gives the F_1 scores of the binary descriptors in two cases (without and with disparity information). It can be seen that, LBP, CLBP, CSLBP and CSLDP with disparity information improve the image retrieval ability as F_1 scores are higher with disparity information than without disparity information. Among them, D-CSLBP is the best one, it achieves the highest F_1 score.

Table 3.1: F_1 score comparison for different tested binary features on four datasets. Here the block size is set to 32×32.

Feature	UTBM-1	UTBM-2	KITTI 05	KITTI 06
LBP	0.5171	0.9058	0.7361	0.8261
D-LBP	0.7665	0.9441	0.7663	0.8639
CLBP	0.5111	0.9194	0.7437	0.8279
D-CLBP	0.6672	0.9221	0.7735	0.8813
CSLBP	0.6292	0.9337	0.7569	0.8461
D-CSLBP	**0.8043**	0.9457	**0.7763**	**0.8850**
CSLDP	0.6093	0.9474	0.7536	0.8335
D-CSLDP	0.8062	**0.9490**	0.7709	0.8709
XCSLBP	0.4796	0.8986	0.7107	0.7814
D-XCSLBP	0.7190	0.8350	0.7401	0.7775

Fig.3-11 depicts the precision-recall curves obtained by the different binary features in two typical datasets UTBM-1 and KITTI 06. It can be seen that, the performance of D-CSLBP is better than the performance with the features D-LBP, D-CLBP, D-CSLDP and D-XCSLBP. Also, it can be seen that the maximum recall at 100% precision for D-CSLBP is higher than the one of the other features.



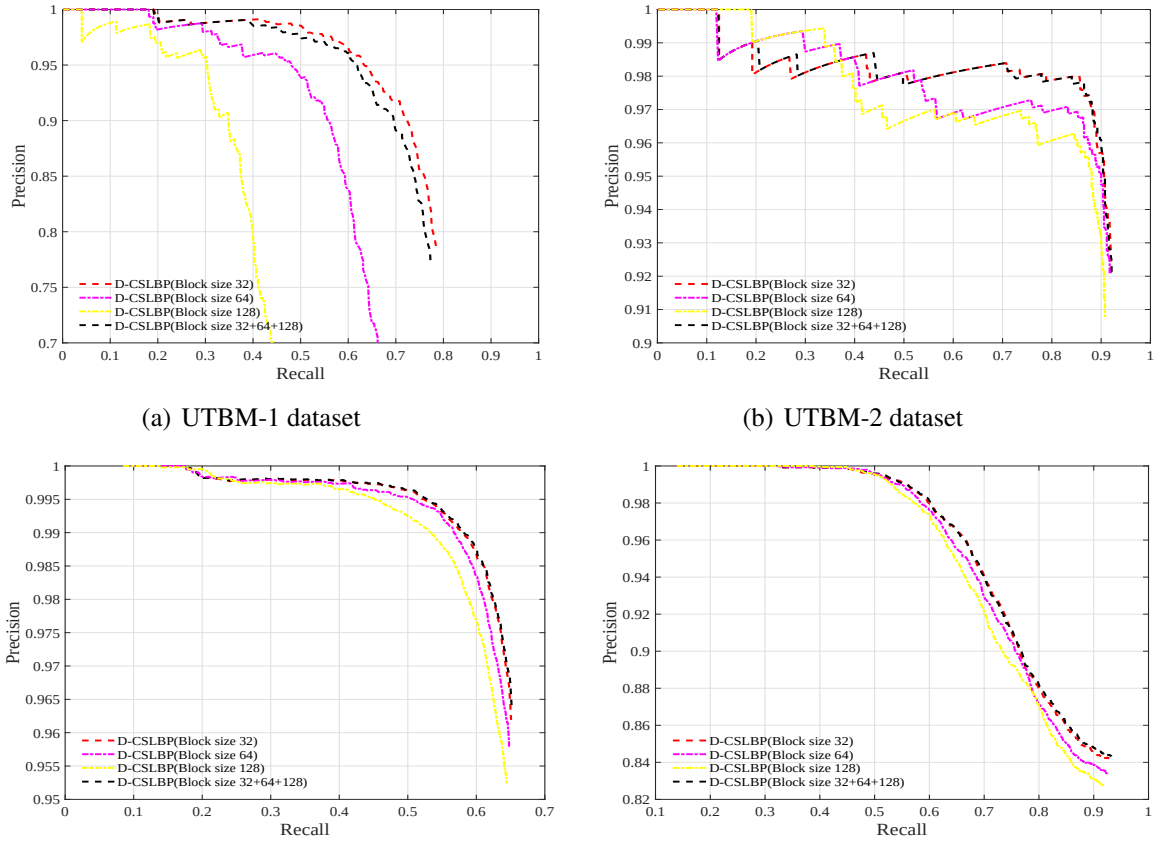
(a) UTBM-1 (b) KITTI 06

Figure 3-11: Image retrieval performance (precision-recall curve) comparison considering different block based binary features on UTBM-1 and KITTI 06 datasets. Here the image block size is 32.

Comparison of different image block sizes

In this section, the influence of block size of block-based D-CSLBP feature is studied.

Small block size permits to discriminate local details, while large block size makes the representation more robust. Each image block is a square block in our experiment (block size 32×32 is short for 32). The performance of D-CSLBP feature with different block sizes (32, 64, 128, and 32+64+128 (multi-block sizes, there different block sizes used together)) in place recognition is evaluated.



(a) UTBM-1 dataset

(b) UTBM-2 dataset

(c) KITTI 05

(d) KITTI 06

Figure 3-12: Image retrieval performance (precision-recall curve) comparison considering D-CSLBP feature extracted with different image block sizes, on four datasets.

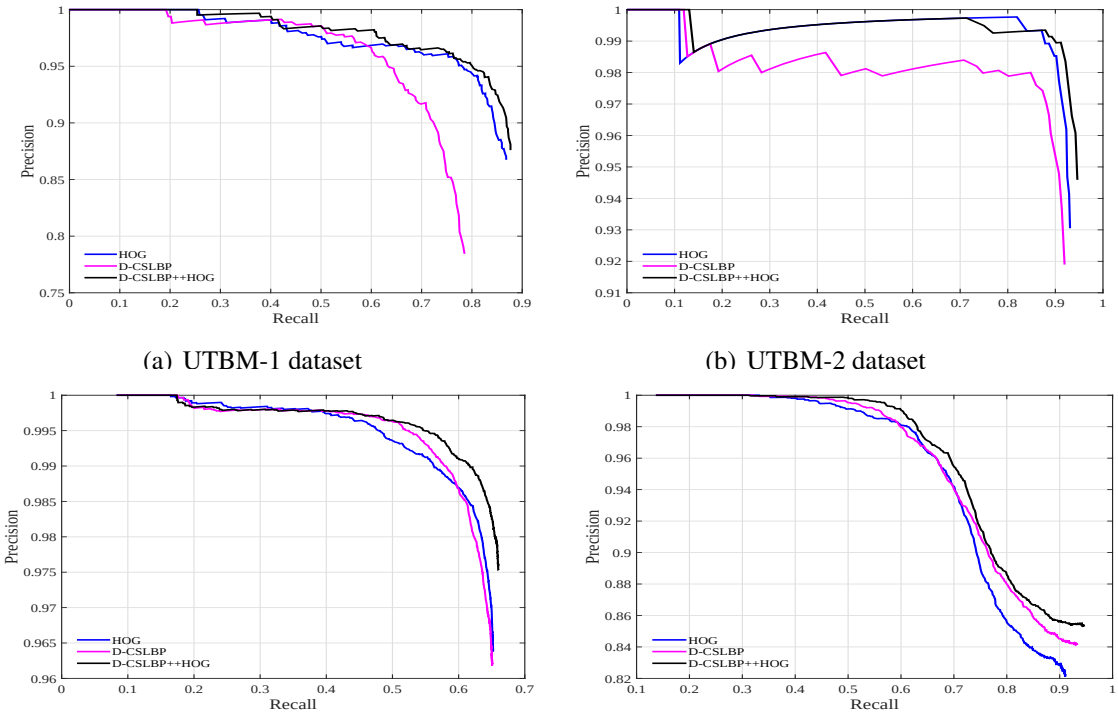
According to Fig.3-12, it can be noted that by increasing the block size from 32 to 64 and 128, the place recognition ability decreases. The computation of D-CSLBP feature with combination of the block sizes 32, 64 and 128 only permits to achieve a slightly better performance than the D-CSLBP feature with block size 32.

It is obvious that, the binary descriptor D-CSLBP extracted from small block size may benefit from discriminative local details. While feature extraction using larger block size makes image representation easy to drop some discriminative information.

However, when the block size is too small, the abundant information can not bring more improvement to the image matching process. At the same time, smaller image block size may lead to computation time increase during feature extraction. So, on our following experiments, the image block size for D-CSLBP is set to 32.

3.5.2 Performance of multi-feature combination

In this section, we compare the performance of multi-feature descriptor (D-CSLBP ++ HOG) with single independent feature descriptor.



(a) UTBM-1 dataset

(b) UTBM-2 dataset

(c) KITTI 05

(d) KITTI 06

Figure 3-13: Image retrieval performance (precision-recall curve) comparison of HOG, D-CSLBP, D-CSLBP++HOG based approaches, on four datasets.

Fig.3-13 shows the precision-recall curves obtained with the different tested features: D-CSLBP, HOG and D-CSLBP ++ HOG. It can be found that the binary feature D-CSLBP

combined with HOG permits to improve image retrieval performance. Combining D-CSLBP and HOG can achieve better result than each single feature, which means that the combination is useful for place recognition.

Table 3.2: Comparison of F_1 scores for different features and the state-of-the-art FAB-MAP method, on four datasets.

Dataset	F_1 score			
	D-CSLBP	HOG	D-CSLBP++HOG	FAB-MAP
UTBM-1	0.8043	0.8752	**0.8869**	0.2356
UTBM-2	0.9299	0.9440	**0.9532**	0.4813
KITTI 05	0.7763	0.7782	**0.7873**	0.7417
KITTI 06	0.8850	0.8648	**0.8973**	0.3519

Table 3.2 compares the F_1 scores of different features with the state-of-the-art FAB-MAP method. It confirms that the multi-feature D-CSLBP++HOG achieves better results than single feature. The F_1 score of D-CSLBP++HOG provides the highest value for all the four datasets. Furthermore, the proposed method outperforms the FAB-MAP method.

For a better comprehension of the proposed multi-feature, an example of distance matrices for UTBM-1 dataset is presented in Fig.3-14. Here, for clearly demonstrates the feature performance, the distance matrix D is normalized into 0-1 range. The distances of same or similar images are close to 0 (red color), while for the larger distances, the corresponding color is close to yellow. As plotted in Fig.3-14 (b), the ground-truth line is a red. When perceptual aliasing occurs, there will be some red points (noisy) appeared which is outside the ground-truth line. In the distance matrix provided by our method using the D-CSLBP++HOG feature (see Fig.3-14 (c)), it can be seen that the noisy which appear around the diagonal (ground-truth line) due to perceptual aliasing are clearly reduced with respect to other feature approaches (CSLBP, D-CSLBP and HOG). All the previous affirmations are supported by the precision-recall curves depicted in in Fig.3-13 (a) and results in Table 3.2.

We can thus conclude that integrating HOG and disparity information permits to improve the image matching results. The reason why the D-CSLBP++HOG achieves better performance than the other features is mainly because the feature combination takes the ad-

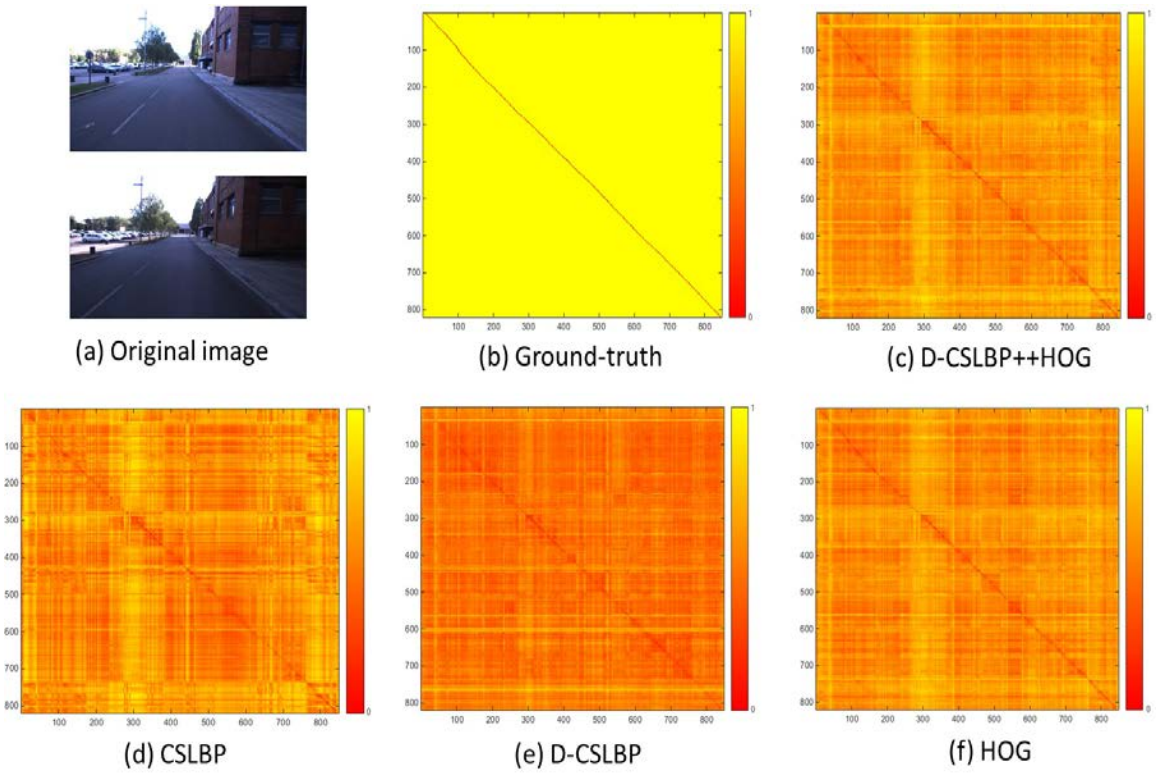


Figure 3-14: Example of distance matrices for UTBM-1 dataset. Here, the distance matrix D is normalized into 0-1 range. The distances of same or similar images are close to 0 (red color), while for the larger distances, the corresponding color is close to yellow. In (a), two images from a same place are taken at different times (difference of two weeks). From figure (b) to figure (f), the distance matrix D is plotted. The distance matrices show that multi-feature combination (c) reduces the noise appeared around the diagonal (ground-truth line). Besides, compared with (d), after adding disparity information in (e), perceptual aliasing decreases, as confirmed by the precision-recall curves in Fig.3-13 (a).

vantage of texture, shape and depth information, which makes image representation more robust than considering each single feature independently.

3.5.3 LSH based visual recognition

Since the block based feature dimension is huge in our approach, computing the euclidean distance between high dimensional feature vectors is an expensive operation. Therefore, in order to speed up image matching significantly, Locality Sensitive Hashing (LSH) method that preserves the euclidean similarity [18], is used for visual recognition. LSH is arguably the most popular unsupervised hashing method and has been applied to many problems,

including information retrieval and computer vision [86]. The paper [86] demonstrates that euclidean distance between two high-dimensional vectors can be closely approximated by the hamming distance between the respective hashed bit vectors. More hash bits that hash method contains, the better approximation is.

The LSH method simply uses a random projection matrix to project the high dimensional data into a low-dimensional binary (Hamming) space; that is, each data point is mapped to a K-bit vector, called the hash key. Thus approximate nearest neighbors in sublinear time can be found. A key ingredient of locality sensitive hashing is mapping similar features to the same bucket with high probability.

More precisely, for multi-features V^{test} obtained from testing image and V^{train} obtained from training image, the hashing functions $H(\cdot)$ from LSH family satisfy the following elegant locality preserving property:

$$P\{H(V^{test}) = H(V^{train})\} = sim(V^{test}, V^{train}) \tag{3.29}$$

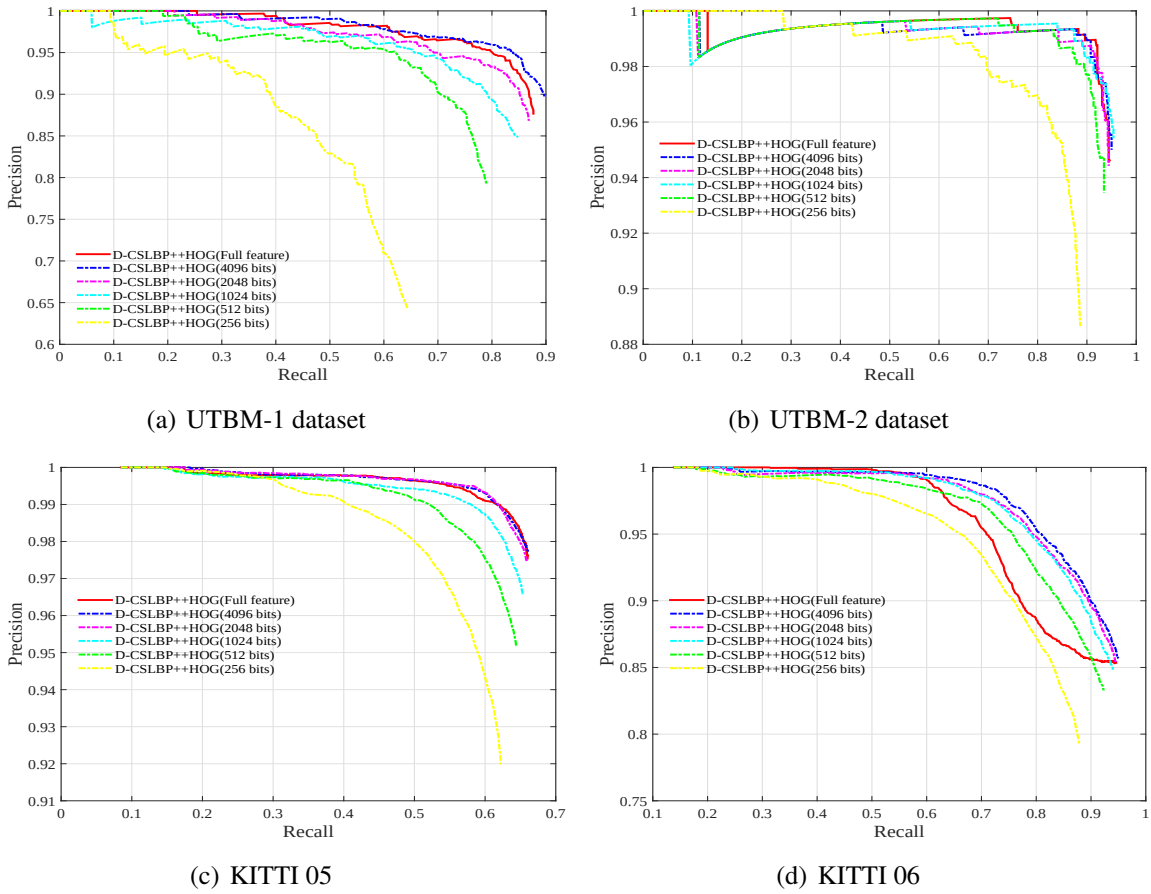
where the similarity measure *sim* is directly linked to the euclidean distance function. Hash keys are constructed by applying K binary-valued hash functions to each image feature. The K binary-valued LSH functions consists of random projections and thresholds as:

$$\begin{aligned} H^{test}(K) &= sign(w^{\top} V^{test} + b) \\ H^{train}(K) &= sign(w^{\top} V^{train} + b) \end{aligned} \tag{3.30}$$

where w is a K dimensional data-independent random hyperplane, which is usually constructed from a standard Gaussian distribution [26]. b is a random intercept. For a normalized data set with zero mean, the approximately balanced partition is obtained with b = 0.

By applying K binary-valued hash functions to each image feature, high dimension multi-features V^{test} and V^{train} are converted into a low K dimension bits H^{test} and H^{train}. Since H^{test} and H^{train} are binary bits, they can be more efficiently compared in low dimension space than original feature.

In our experiment, we compare the place recognition performance achieved with hashed



(a) UTBM-1 dataset

(b) UTBM-2 dataset

(c) KITTI 05

(d) KITTI 06

Figure 3-15: Image retrieval performance (precision-recall curve) comparison of different hash bit lengths.

multi-feature of different binary lengths ($2^8 \ldots 2^{12}$ bits) on four datasets in Fig.3-15. Since the image size is different, multi-feature dimension in datasets UTBM-1 and UTBM-2 is 18696 while the multi-feature dimension in KITTI 05 and KITTI 06 is 6432. It can be seen that, using 4096 and 2048 bits retain above 86% total place recognition performance.

Table 3.3 shows the F_1 score obtained form different hash bit lengths applied on the multi-feature (D-CSLBP++HOG) of our place recognition method. The average matching time is also presented. Here average matching time does not include the feature extraction time. The experiments were conducted on a laptop machine with intel i7-4700MQ CPU and 32G RAM.

As Table 3.3 shows, the average matching time using 4096 bits is almost half of the one using the euclidean distance over the original full features. Compared with the full multi-feature matching, hashing the original multi-feature into 4096 bits makes the dis-

tance computation and comparison easier and faster. There is no doubt that, for large scale datasets, the speed up advantages can be more significant.

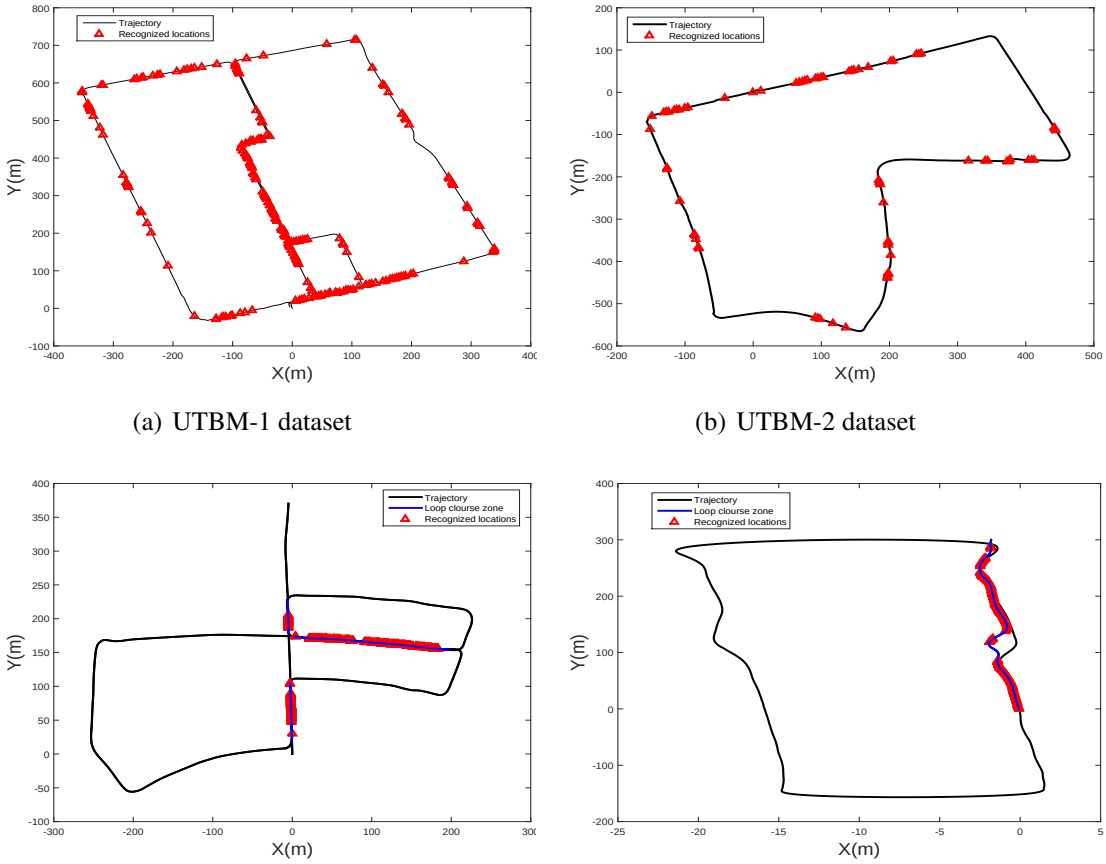
Table 3.3: F_1 score and matching times comparison of different hash bit lengths for our approach and the state-of-the-art FAB-MAP method.

Method	F_1 score				Average time per matching (All datasets)
	UTBM-1	UTBM-2	KITTI 05	KITTI 06	
256 hash bits	0.6571	0.8989	0.7425	0.8411	0.11×10^{-2} s
512 hash bits	0.8097	0.9409	0.7862	0.8786	0.38×10^{-2} s
1024 hash bits	0.8488	0.9562	0.7794	0.8936	0.88×10^{-2} s
2048 hash bits	0.8771	0.9532	0.7865	0.8973	1.78×10^{-2} s
4096 hash bits	0.8869	0.9537	0.7873	0.9006	3.61×10^{-2} s
Multi-feature	**0.9258**	0.9110	**0.9166**	**0.9203**	8.82×10^{-2} s
FAB-MAP	0.2356	0.4813	0.7417	0.3519	2.83×10^{-2}s

3.5.4 Visual localization results

In the previous section, 4096 bits obtained by hashing the original feature shows its good performance in place recognition. Therefore, in this section, we describe visual localization results achieved by 4096 hash bits. Fig.3-16 shows the final place recognition results for the different datasets at a precision level of 100%. For the datasets UTBM-1 and UTBM-2, we obtained 23.81% and 11.35% recall at the 100% precision respectively. While in the KITTI 05 and KITTI 06 datasets, a recall rate of 17.38% and 32.39% is achieved respectively at the total correctly level. It should be noted that, at 100% precision level, the obtained place recognition result is totally correct. A correct place recognition means a successful visual localization, therefore, high recognition rate (recall) at 100% precision is, more robust visual localization system is.

Fig.3-17 shows the place recognition based visual localization errors at different precision levels. When adjusting the threshold value Th, the recognition precision is also changing. At 100% precision level, each recognized place is true positive and its localization error is small (depending on the ground-truth criteria, in our case it is 5m). For achieving the 100% recognition precision level, threshold value is set to 0.88 and 0.58 for



(a) UTBM-1 dataset

(b) UTBM-2 dataset

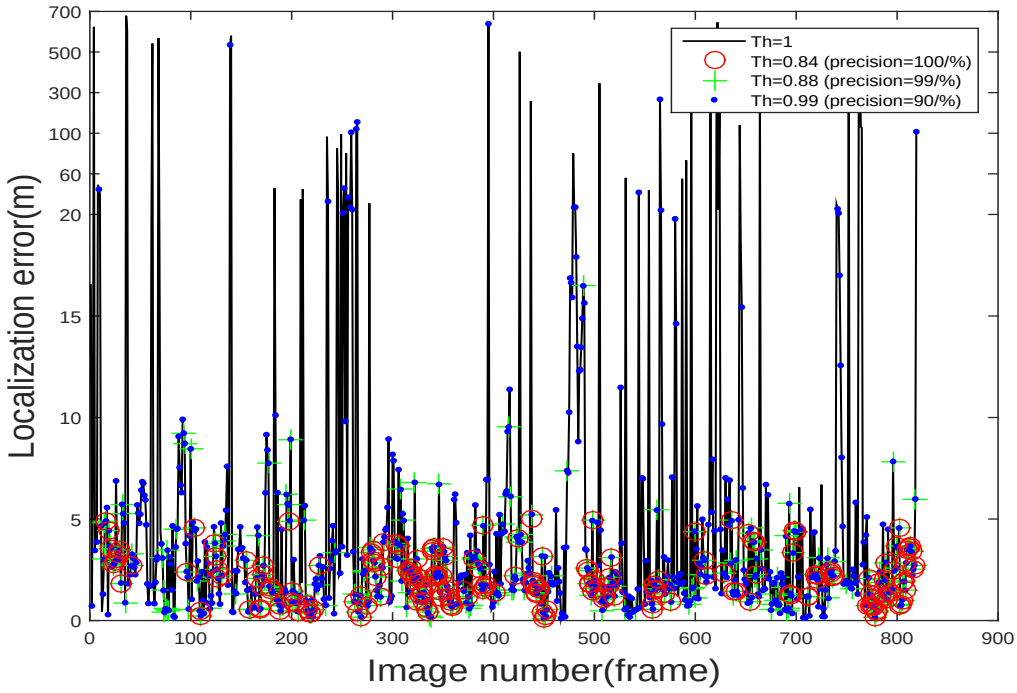
(c) KITTI 05

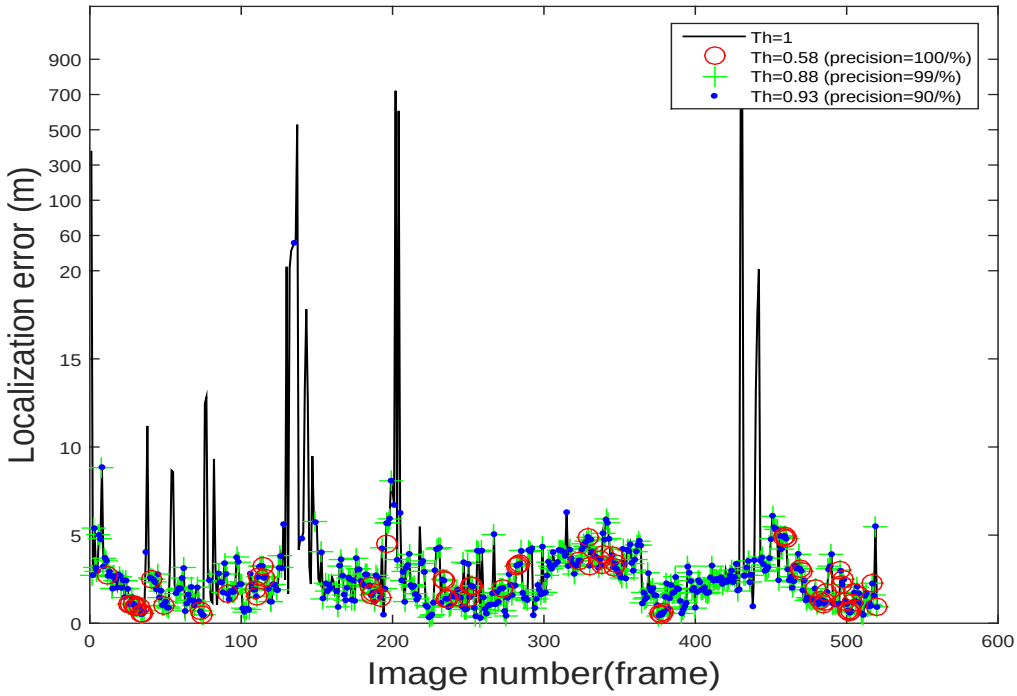
(d) KITTI 06

Figure 3-16: Visual localization results obtained by our system on four datasets. The trajectory of the vehicle is depicted with black lines, the loop closure zone is plotted by blue lines. Red points are correctly recognized locations at 100% precision by using our proposed approach. There are no false positives in any case. It is noted that the loop closure zone of datasets UTBM-1 and UTBM-2 is the whole trajectory, while the loop closure zone of KITTI 05 and KITTI 06 is only parts of the trajectory in blue.

UTBM-1 and UTBM-2 datasets respectively. It can be seen that, the visual localization error is below 5m at 100% precision level. While with threshold Th increased to 0.9, the precision level is decreasing and localization error of some locations exceeds 5m . When the threshold is set to 1, which means every image matching result is positive, in this case, the precision level is lowest and there are many false matching for place recognition, which lead to huge localization error. In general, if small threshold is used, there are few false recognition cases.

In addition, for visual recognition precision level below 100% , meaning that recog-





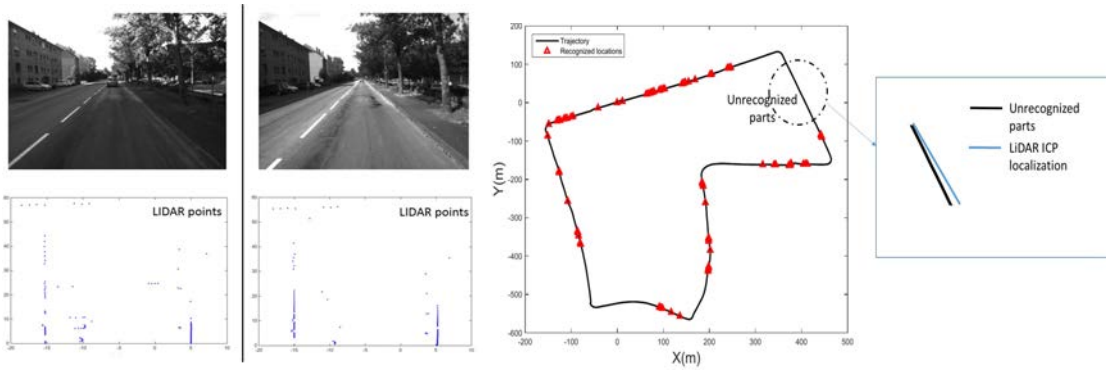
(b) UTBM-2 dataset

Figure 3-17: Place recognition based localization errors at different precision levels. The dark line is the localization error for all recognition results (including the true positives and false positives). Red "o" is localization error at 100% precision level. Green "+" is localization error at 99% precision level. Blue "." is localization error at 90% precision level.

nized places are not totally correct, some false recognition places appear. For these false recognized places, the localization error can be very large, because the testing image can be wrongly matched to anyone in the training image database. That is also the reason why some locations have huge localization error.

Table 3.4: Recall results and average localization error at three precision levels (4096 hash bits).

Dateset	100% precision		99% precision		90% precision	
	Recall (%)	Error (/m)	Recall (%)	Error (/m)	Recall(%)	Error (/m)
UTBM-1	23.81	2.08	89.62	2.34	95.00	3.53
UTBM-2	11.35	2.11	51.89	2.41	89.50	2.49
KITTI 05	17.38	3.50	61.59	13.08	65.92	13.08
KITTI 06	32.39	2.56	59.12	3.18	77.26	4.20



(a) Example of two images and their corresponding LiDAR data.

(b) LiDAR based localization results. It can be seen that using LiDAR only for long-term localization, will lead to accumulated errror.

Figure 3-18: An example of LiDAR localization results.

Table 3.4 gives the average localization error and recall ratio at different precision levels. For all these datasets, at 100% precision, the minimum localization error is 0 while the maximum error is not larger than 5 m. It should be noted that, at 100% precision level, some places can not be recognized and no localization results are obtained at these places. This problem can be easily solved by visual odometry technique or extra sensors (as LiDAR or Inertial Measurement Unit (IMU)). Fig.3-18 shows the example of LiDAR based localization in case of unrecognized positions with our place recognition based method.

Here, a 2D LiDAR is used, and the transformation between two successive LiDAR point sets can be computed using ICP (Iterative Closest Point) method. Thus, based on the previous position and the transformation information, the positions of unrecognized places can be computed.

3.6 Conclusion and Future Works

In this chapter, we presented a visual vehicle localization approach that uses multi-feature built from gray-scale image and disparity map. The multi-feature concatenates the D-CSLBP and HOG features together to take the advantage of texture, depth and shape information. Also, block based feature extraction was used to consider the spatial information. Image matching using the proposed multi-feature D-CSLBP++HOG based on local sensitive hashing makes the visual recognition more efficient. The results of our experiment demonstrated that this approach provides an available place recognition based visual localization in outdoor environment compared with the state of art FAB-MAP method.

However, in the long-term visual localization, place recognition is prone to be influenced by appearance or seasonal changing. The future objective of our research is to achieve a robust long-life localization at different times and seasons. Sequence matching will be considered for place recognition in the following research.

Chapter 4

Visual Localization Across Seasons Using Sequence Matching Based on Feature Combination

In Chapter 3, visual localization is achieved by single image matching based on multi-feature D-CSLBP++HOG. The multi-feature obtained from gray-scale image and disparity map showed its advantages in place describing. However, using stereo camera to get disparity map, increases the hardware costs and processing time. In addition, HOG feature is easy to be influenced by huge appearance variation, different illumination and seasonal changing in long-term localization.

In this chapter, the problem of visual localization across seasons is addressed using feature combination and sequence matching. Feature combination of CSLBP and GIST is used to make place describing more robust, and sequence matching rather than single image matching is used to improve the place recognition ability. Based on the above consideration, a more robust visual localization system based on feature combination and sequence matching is proposed. Experimental evaluation will show that our method is an effective tool to perform visual localization across seasons. The results will also show an improved precision-recall performance against state-of-the-art SeqSLAM algorithm.

4.1 Introduction

Visual localization is an ongoing challenge in robotics, especially in seasonal changing situation. A single can look extremely different depending on the current season and weather conditions. Visual localization system across different seasons must be robust without being influenced by seasonal and weather variations that lead to vast variations in image appearance.

As already cited in the previous chapter, FAB-MAP is a single image based matching algorithm, which employs Bag-of-Words (BOW) image retrieval technique and a Bayesian frame-work [24] to achieve robust place recognition. Recently, sequence SLAM (SeqSLAM) [72] adopting sequence matching rather than single image matching for place recognition achieves significant performance improvements with respect to FAB-MAP. The usage of sequences allows higher robustness to lighting or extreme perceptual changes. In SeqSLAM, image similarity is evaluated using the sum of absolute differences between contrast enhanced and low-resolution images without the need of image keypoint extraction. However, in [100] some weaknesses of SeqSLAM were reported, such as the field of view dependence and the complexity of parameters configuration. For these reasons, the community continues searching for new methods which can satisfy the high requirements needed to achieve robust life-long visual localization.

Besides that, CSLBP is one of the widely used binary descriptors, which is invariant to monotonic changes in gray-scale and fast to calculate. As proved in Chapter 3, CSLBP has strong place describing ability. Considering object shape information is changing (e.g. leaves falling down, snow covering) at different seasons, GIST feature rather than HOG is used in this work. GIST focuses more on the whole scene itself and on the relationship between the outlines of the surfaces and their properties [30]. GIST and CSLBP can be seen complementary for image representation in the sense that GIST focuses more on global information and CSLBP emphasizes local texture information. Inspired by the advantages of feature combination, CSLBP and GIST are combined together for place describing in this chapter.

In this chapter, we present a visual localization method using sequence matching based

on feature combination that is robust to extreme perceptual changes. Fig.4-1 illustrates the general diagram of our approach. Based on features extracted from images, sequences are efficiently matched using Chi-square distance and the best candidate to place matching is recognized according to coherent sequence matching. Thus, visual localization is realized through the recognized places. Image feature used in this chapter is a combination of CSLBP and GIST, which should improve image distinguishing ability by capturing local and global image information. We will demonstrate the algorithm performance using multi-season videos of 30000 km long train ride in the northern Norway. For this, an extensive experimental study is conducted according to sequence matching length, as well as a comparison of the proposed approach with the state of the art SeqSLAM [72] method.

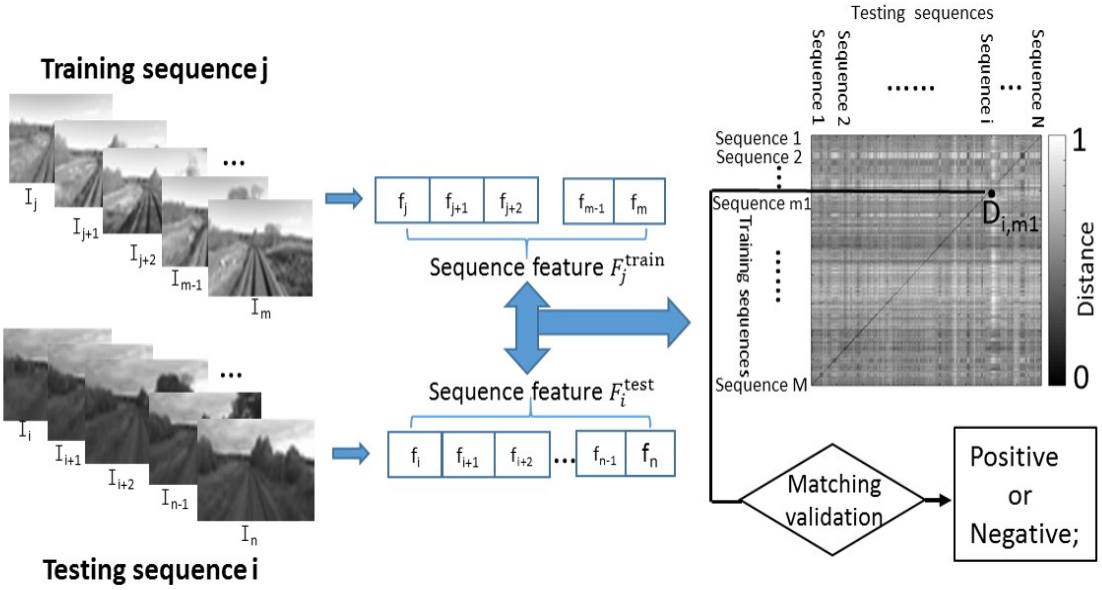


Figure 4-1: General diagram of visual localization system using sequence matching.

This chapter is organized as follows: Section 4.2 describes the proposed visual localization approach. Section 4.3 details experiment setup: the used dataset and evaluation method. Experiments are presented with results in Section 4.4. Finally, Section 4.5 discusses the outcomes of this chapter and presents future work.

4.2 Proposed Visual Localization Approach

The proposed visual localization method using CSLBP and GIST features to represent image sequence, is realized based on sequence matching. As illustrated in Fig.4-1 and Fig.4-2, there are five main components in our approach: image preprocessing, feature extraction, sequence matching, matching validation and the last is visual localization based on the matching result.

To detail, a set of GPS tagged training images is firstly acquired. After image preprocessing (Section 4.2.1), CSLBP and GIST features are extracted independently from the images of the training database and then concatenated together to form multi-feature CSLBP++GIST (Section 4.2.2). Then, multi-feature CSLBP++GIST obtained from images of a sequence are concatenated (++) to form the final sequence feature (F) representing the sequence. Here, sequence consists of consecutive images and each sequence is independent. A current place (represented by a testing sequence) is then recognized through

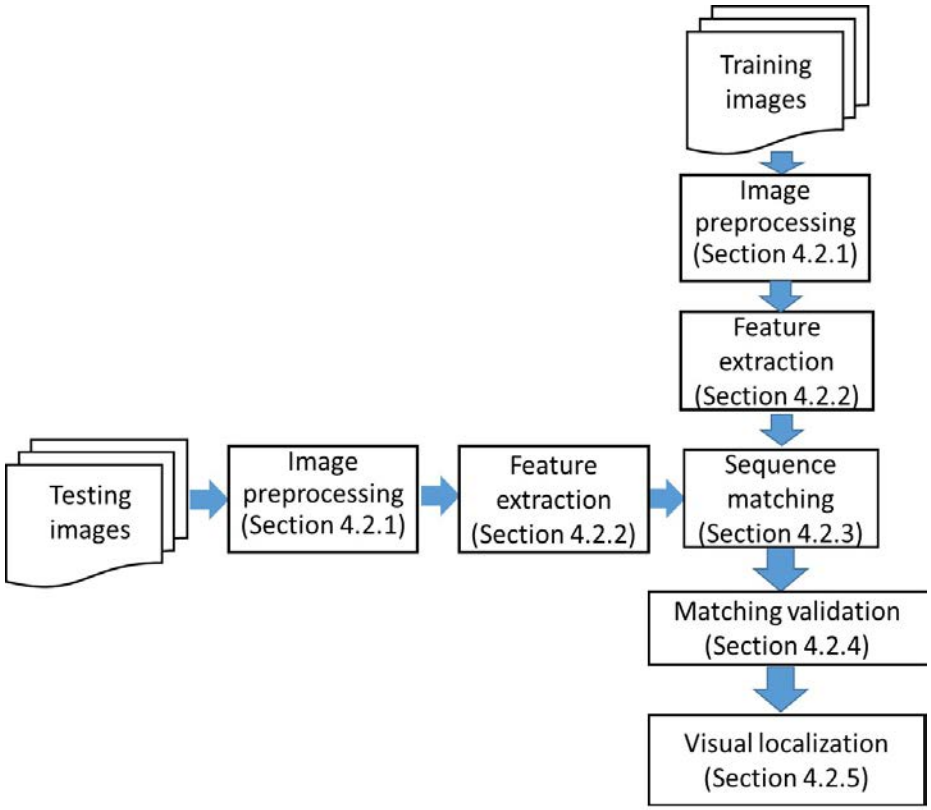


Figure 4-2: Flow chart of proposed visual localization using sequence matching.

sequence matching step based on Chi-square distance (Section 4.2.3).

In the step of sequence matching, for each testing sequence, the sequence candidate from the training database that provides the minimum distance can be considered as the most similar one to the testing sequence. In fact, the two best sequence matching candidates are conserved for further verifying the final matching result.

Effectively, the best matching candidate will be validated through a distance ratio SS (Section 4.2.4), computed from the two minimum scores (the first minimum distance divided by the second minimum distance). If the ratio SS is lower than or equal to a threshold Th, the first best sequence candidate (with the lower distance) is confirmed and regarded as positive matching, otherwise it is considered as negative one (in this case, no matching result is conserved). When a sequence candidate is confirmed as positive, the position can be obtained from the GPS information that correspond to the matched training images (Section 4.2.5).

4.2.1 Image preprocessing

As mentioned in paper [89] and already considered in the previous chapter, high resolution images are not needed to perform an effective visual recognition along time. Indeed, high resolution images increase computational cost without bringing significant visual recognition improvement. For image storage and efficient matching, in this work, the original images are down-sampled into 32×32 pixels before feature extraction.

4.2.2 Feature extraction

Feature extraction consists of three steps: (1) CSLBP and GIST are firstly extracted from image independently; (2) Then they are concatenated (++) together to form multi-feature; (3) Finally, the CSLBP++GIST multi-features obtained from images of a sequence are concatenated (++) to form the final sequence feature (F) representing the sequence.

1) CSLBP feature: As already described, CSLBP is a modified version of LBP. We recall that, for an even number P of neighboring pixels distributed on radius R, the feature of uniform LBP pattern (the property of the "uniform patterns" is that the number of 0-1

transitions is no more than 2) is a $P(P-1)+3$ dimension histogram (details can be found in Section 3.2 of Chapter 3). While CSLBP operator produces $2^{P/2}$ patterns as follows:

$$CSLBP_{P,R}(x_c, y_c) = \sum_{i=0}^{(P/2)-1} s(|g_i - g_{i+(P/2)}|)2^i \tag{4.1}$$

$$s(x) \begin{cases} 1, & x > T \\ 0, & otherwise \end{cases} \tag{4.2}$$

where g_i and $g_{i+(P/2)}$ correspond to the gray values of center-symmetric pairs of pixels (P in total) equally spaced around central pixel (x_c, y_c). T is used to threshold the gray-level difference so as to increase robustness of CSLBP feature on flat image regions. Since the gray levels are normalized in [0,1], the authors of paper [43] recommend to use small value for T.

CSLBP is closely related to the gradient operator, because it compares the gray levels of pairs of pixels in centered symmetric directions instead of comparing the central pixel to its neighbors. In this way, CSLBP feature takes advantage of the properties of both LBP and gradient based features. For an image of size 32×32, after CSLBP pattern of each pixel is computed, a histogram is built to represent the image texture:

$$f_{CSLBP} = \sum_{i=1}^{32} \sum_{j=1}^{32} f\left(CSLBP_{P,R}(i,j), l\right), \; l = 0,1,2,3,\cdots,2^{P/2}-1, \tag{4.3}$$

$$f(x,y) = \begin{cases} 1, & x = y \\ 0, & othersize \end{cases} \tag{4.4}$$

By construction, the length of the histogram resulting from the CSLBP feature is $2^{P/2}$. It is obvious that CSLBP produces shorter feature set than LBP. Also, it is a first order local pattern in center symmetric direction and it ignores the central pixel information.

In this work, 8 sampling points and 3 pixels radius around the center pixel are set, thus 16-dimensional CSLBP features are obtained.

2) GIST feature: It is a global image feature, which characterizes several important statistic information about a scene [94]. A variety of experimental studies have demon-

strated that humans perform rapid categorization of a scene by integrating only coarse global information that can be extracted using " GIST " [97]. Using the model proposed by Oliva [81], GIST feature is computed by convolving an oriented filter with down-sampled images (32×32) at several different orientations and scales. The scores for the filter convolution at each orientation and scale are stored in an array and resulting in a 512-dimensional feature.

3) Feature combination: After getting CSLBP feature f_{LBP} and GIST feature f_{GIST} from an image, they are combined into a new CSLBP++GIST feature f. The combination consists simply in concatenating (++) the two features:

$$f = f_{CSLBP} + + f_{GIST} \tag{4.5}$$

Combining CSLBP and GIST features allows taking simultaneously advantage of local and global image information and thus allows representing the scene of each location more comprehensively.

Fig.4-3 illustrates an example of extracted features. It can be seen that CSLBP and LBP features pay more attention to the image detail (local information), and CSLBP represents better the image than LBP. While GIST feature pays more attention to the whole scene. Therefore, combination of the local feature CSLBP and global feature GIST will describe the place better and should improve place recognition performance.

4) Sequence feature: Finally, the CSLBP++GIST features extracted from images of a sequence are concatenated (++) to form the final sequence feature (F) representing the sequence of images:

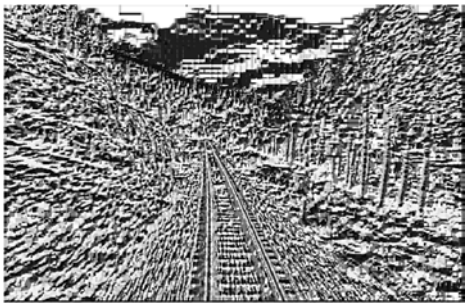
$$F = f_i + + f_{i+1} + + f_{i+2} + + ... + + f_{m-2} + + f_{m-1} + + f_m \tag{4.6}$$

where, i, $i+1$, $\cdots$ m are the indexes of the consecutive images composing the sequence, and $L_{length} = m - i + 1$ is the length of the sequence. The total feature dimension is $528 \times L_{length}$. 528 is the sum of 16 (dimension of CSLBP feature) and 512 (dimension of GIST feature).

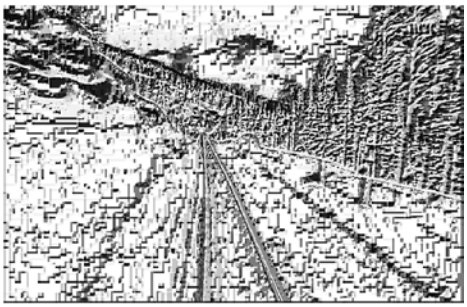
Here, each sequence is composed of consecutive images. For that, original image

(a) Summer image

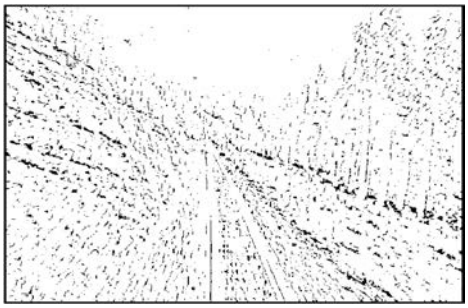
(e) Winter image



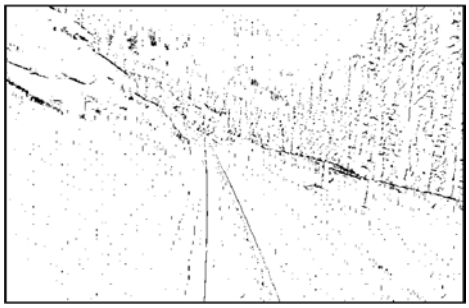
(b) LBP image



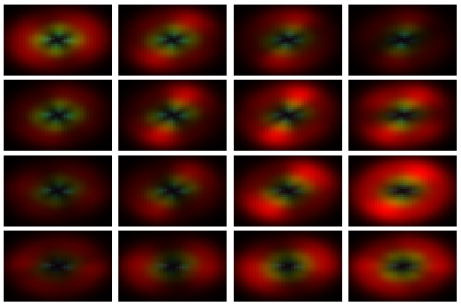
(f) LBP image



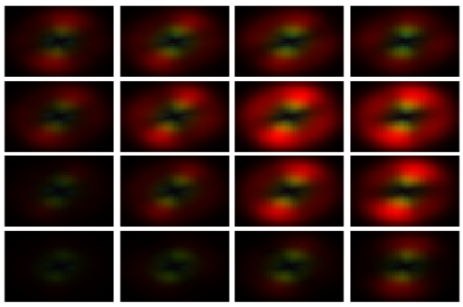
(c) CSLBP image



(g) CSLBP image



(d) GIST image



(h) GIST image

Figure 4-3: Example of extracted features. The first row shows the original images. The second and third rows show the images of LBP and CSLBP features respectively. The fourth row gives the images of GIST features.

database is simply divided into sequences with the same length. Since training and testing route is traveled with a similar speed, the same sequence length is also used for testing sequence. Thus, each sequence can be represented using sequence feature (F) with the same dimension.

4.2.3 Image sequence matching

To perform sequence matching, similarity between sequence features is evaluated through Chi-squared distance. The Chi-squared distance is a nonlinear metric which can be calculated easily. Suppose the numbers of training and testing images are N^{train} and N^{test} respectively, and sequence length is L_{length}. Therefore, the training sequence number is $M = N^{train}/L_{length}$ and the testing sequence number is $N = N^{test}/L_{length}$.

Given a testing sequence Q_i^{test} (composed of the L_{length} last consecutive testing images), it will be compared with each training sequence $Q_j^{train}(j = 1,2,\cdots,M)$ from the training database. Since the sequence lengths are same, therefore the sequence matching can be conveniently conducted using the Chi-squared distance.

The similarity value between the two sequences Q_i^{test} and Q_j^{train} is measured using the Chi-squared distance D_{ij}, computed as follows:

$$D_{i,j} = \chi^2(F_i^{test}, F_j^{train}) = \sum_k \frac{((F_i^{test})_k - (F_j^{train})_k)^2}{|(F_i^{test})_k + (F_j^{train})_k|} \tag{4.7}$$

Where F_i^{test} is the feature vector of the current testing sequence Q_i^{test}, F_j^{train} is the feature vector of a training sequence Q_j^{train} (from the training dataset). k is index of the components of feature vector: $k = 1,2,\cdots,528 \times L_{length}$. Then, all the computed $D_{i,j}$ form a distance matrix D.

For sequence matching, feature vector of the current sequence Q_i^{test} is compared with feature vector of each training sequence $Q_j^{train}(j = 1,2,\cdots,M)$. Based on the distances $D_{i,j}(j = 1,2,\cdots,M)$, the two best training sequence candidates (which have the first minimum distance and second minimum distance) that best match the current testing sequence are conserved.

4.2.4 Matching validation

In order to reduce false matching cases, the ratio SS_i computed from the first minimum distance and second minimum distance is used to validate the matching result (as for single image matching):

$$SS_i = \frac{D_{i,m1}}{D_{i,m2}} \tag{4.8}$$

where $D_{i,m1}$ and $D_{i,m2}$ are respectively the first and second minimum distances between the feature vector of current testing sequence Q_i^{test} and the feature vectors of all the training sequences $Q_j^{train}(j = 1,2,\cdots,M)$, as follows:

$$\begin{cases} D_{i,m1} = \min\limits_{j}\{D_{i,j}\} \\ D_{i,m2} = \min\limits_{j(j\neq m1)}\{D_{i,j}\} \end{cases} \tag{4.9}$$

The values of SS_i are between 0 and 1. A threshold is then applied to the score SS_i to determine if the sequence pair $(i,m1)$ is matched or not. The matching is considered as positive when the distance ratio SS_i is lower than or equal to the threshold Th, otherwise it is considered as negative and the sequence pair is ignored.

4.2.5 Visual localization

After one sequence matching candidate is successfully validated, the vehicle can localize itself through the GPS information attached to the matched training sequence. Effectively, since the training images are tagged with GPS or pose information, the vehicle can get its position through the training images that matched with the current testing sequence. This is a topological level localization, that is, the system simply identifies the most likely location. Therefore, this is not a very accurate localization, because the training and testing trajectories are not exactly same.

4.2.6 Algorithm of proposed visual localization

Algorithm 4.1 illustrates the proposed method for sequence matching based visual localization. It includes feature extraction and combination, image sequence matching, matching validation and visual localization steps.

Algorithm 4.1 Sequence matching based visual localization using feature combination.

Inputs:

$\{I_j^{train}\}_{j=1}^{N^{train}}$ {training images}; $\{I_i^{test}\}_{i=1}^{N^{test}}$ {testing images };
N^{train},N^{test} {training and testing images numbers};

Outputs:

SS{distance ratio}; Vehicle position

Algorithm:

/* Feature extraction and combination; {Section 4.2.2} */
for $jj \leftarrow 1$ to N^{train} **do**
  f_{CSLBP},$f_{GIST} \leftarrow$ CSLBP and GIST features extraction for training images;
  $f_{jj}^{train} \leftarrow f_{CSLBP} + + f_{GIST}$; //Feature combination.
end for
for $j \leftarrow 1$ to N^{train}/L_{length} **do**
  $jj \leftarrow (j-1) \times L_{length}$;
  $F_j^{train} \leftarrow f_{jj+1}^{train} + + f_{jj+2}^{train} + + \cdots f_{jj+L_{length}}^{train}$; //Feature of training sequence.
end for

for $ii \leftarrow 1$ to N^{test} **do**
  f_{CSLBP},$f_{GIST} \leftarrow$ CSLBP and GIST features extraction for testing images;
  $f_{ii}^{test} \leftarrow f_{CSLBP} + + f_{GIST}$; //Feature combination.
end for
for $i \leftarrow 1$ to N^{test}/L_{length} **do**
  $ii \leftarrow (i-1) \times L_{length}$;
  $F_i^{test} \leftarrow f_{ii+1}^{test} + + f_{ii+2}^{test} + + \cdots f_{ii+L_{length}}^{test}$; //Feature of testing sequence.
end for

/* Sequence matching based on feature sequences; {Section 4.2.3} */
for $i \leftarrow 1$ to N^{test}/L_{length} **do**
  for $j \leftarrow 1$ to N^{train}/L_{length} **do**
    $D_{i,j} \leftarrow \sum_k \frac{((F_i^{test})_k - (F_j^{train})_k)^2}{|(F_i^{test})_k + (F_j^{train})_k|}$; //Chi-square distance computation, k is the index of the compents of feature vector.
  end for

  /* Matching validation and visual localization; {Section 4.2.4 and 4.2.5} */
  $SS_i = \frac{\min_j \{D_{i,j}\}}{\min_{j(j \neq m1)} \{D_{i,j}\}}$; $m1$ is the index of the first minimum distance.
  if $SS_i <= Th$
    Matching validation is positive;
    Vehicle position $\leftarrow$ the matched training image position
  if $SS_i > Th$
    Matching validation is negative;
    Vehicle position $\leftarrow$ NaN (no position result)
end for

4.3 Experimental Setup

In this section, the used dataset is described as well as ground-truth and image pre-processing.

4.3.1 Dataset and ground-truth

The dataset used in our work is an open dataset called Nordland[1]. It is composed of footage videos of a 728 km long train ride between two cities in north Norway [100] (see Fig.4-4). The complete 10 hours journey has been recorded in four seasons. Thus, the dataset can be considered as a single 728 km long loop that is traversed four times. As illustrated in Fig.4-5, there is an immense variation in the appearance of the landscape, reaching from green vegetation in spring and summer to colored foliage in autumn and complete snow-cover in winter over fresh. In addition to the seasonal changes, different local weather conditions

Figure 4-4: Nordland route. Source: Google map. The trajectory is recorded four times, once in every season. Video sequences are synchronized and the camera position and field of view are always the same. GPS readings are available.

[1]https://nrkbeta.no/2013/01/15/nordlandsbanen-minute-by-minute-season-by-season/

like sunshine, overcast skies, rain and snowfall are experienced on the long trip. Most of the journey leads through natural scenery, but the train also passes through urban areas along the way and occasionally stops at train stations or signals. The original videos have been recorded at 25 fps with a resolution of 1920×1080 using a SonyXDcam with a Canon image stabilizing lens. GPS readings were recorded in conjunction with the video at 1 Hz. The full-HD recordings have been time-synchronized such that the position of the train in an arbitrary frame from one video corresponds to the same frame in any of the other three videos. This was achieved by using the recorded GPS positions through interpolation of the GPS measurements to 25 Hz to match the video frame rate.

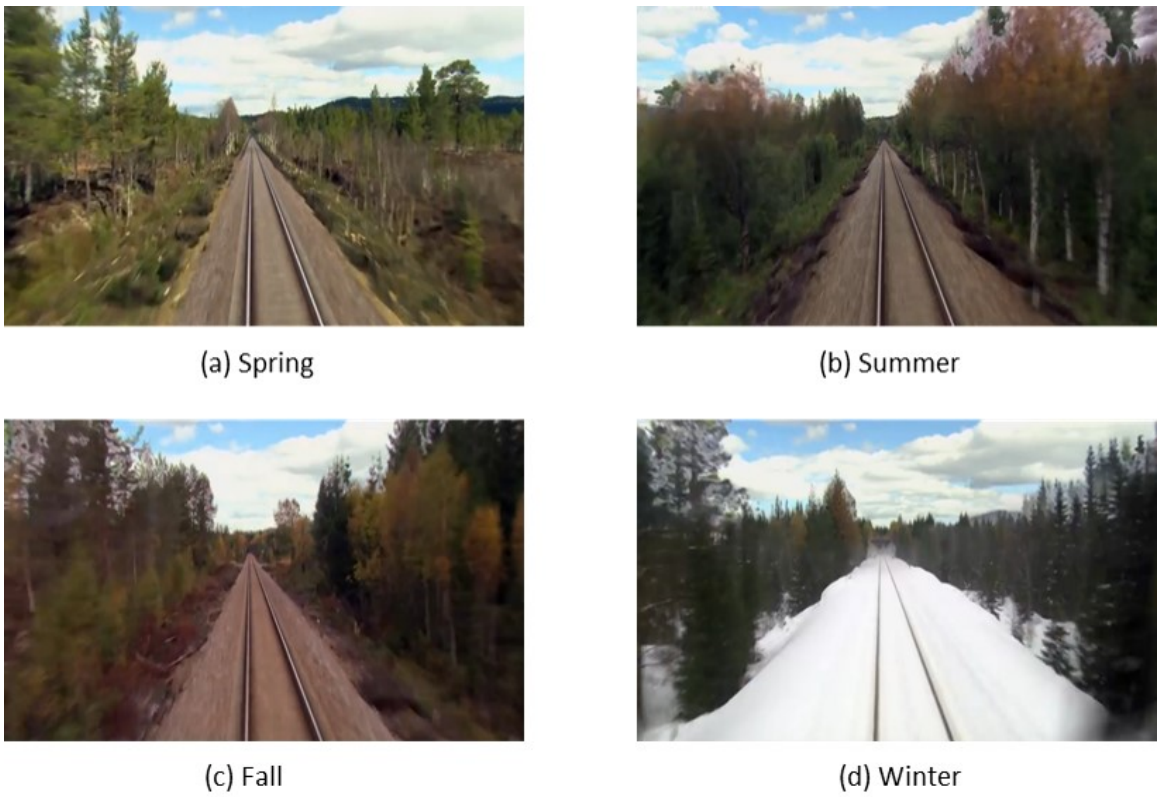


Figure 4-5: A typical four seasons images representing the same scene in spring, summer, fall and winter. It can be seen that huge differences appear in the images with season changing.

For the experiments described in the following, image frames are extracted from the original videos at 1 fps, there are then 35768 image frames for each season. Each image is then down-sampled these images to 32×32 pixels and converted into gray-level images.

4.3.2 Evaluation method

Precision-recall characteristics are widely used to evaluate the effectiveness of image retrieval. Therefore, as used in the previous chapter, our evaluation methodology is based on precision-recall curves. These curves are determined by varying the threshold Th between 0 and 1, applied to the ratio SS and calculating precision and recall (see section 4.3.2). Precision relates to the number of correct matches to the number of false matches, whereas recall relates to the number of correct matches to the number of missed matches. Positives are considered when ratio is lower than or equal to the threshold Th. Here 100 threshold values are considered to obtain well-defined precision-recall curves.

In this experiment, the training image number is equal to the testing image number, and each testing image has a ground truth matching. Therefore, there are only true positives (correct results among successfully validated image matching candidates) and false positives (wrong results among successfully validated image matching candidates). The sum of the true positives and false positives is the total retrieved image numbers.

4.4 Experiments and Results

4.4.1 Feature combination analysis

In a first set of experiments, we evaluate how well do feature combinations perform for place recognition and also compare the results with those obtained by the state-of-art SeqSLAM method. The experiments were conducted using the videos presenting extreme situation in terms of appearance changes (Spring vs Winter). The length of each sequence is 200 images. As shown in Fig.4-6, the CSLBP and GIST features perform very well when they are used independently. Indeed, our method with CSLBP performs relatively well at high precision level, while GIST outperforms SeqSLAM method. When using the multi-feature (CSLBP++GIST), the retrieval ability is increased significantly. The reason is that CSLBP++GIST takes advantage of local and global information can distinguish the similar images more accurately.

It can be seen that our method with CSLBP++GIST can reach around 65% of recall at

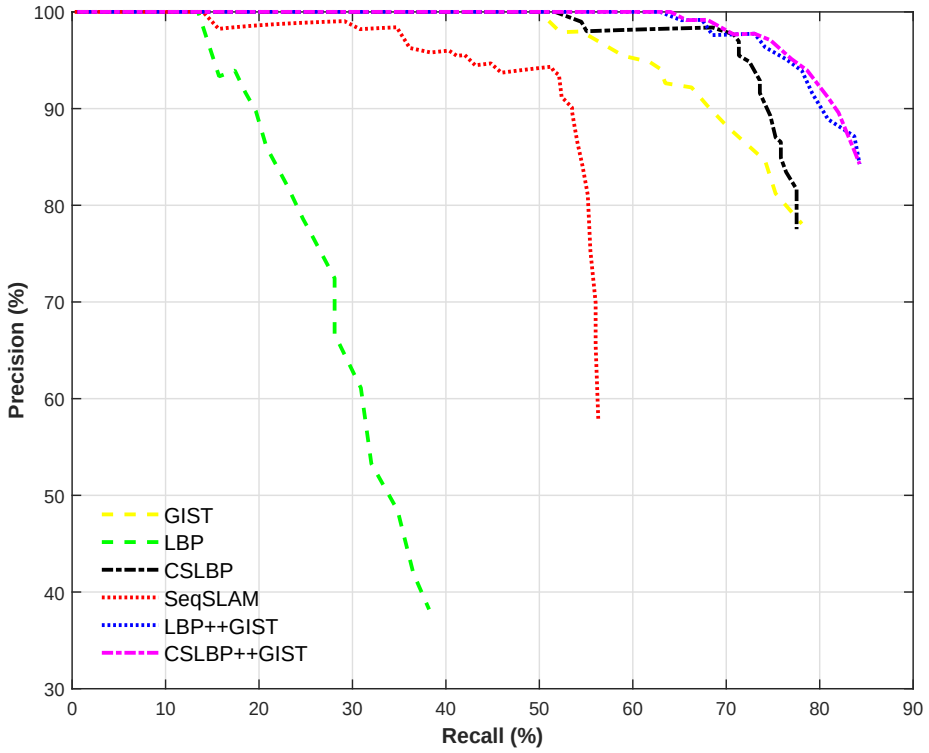


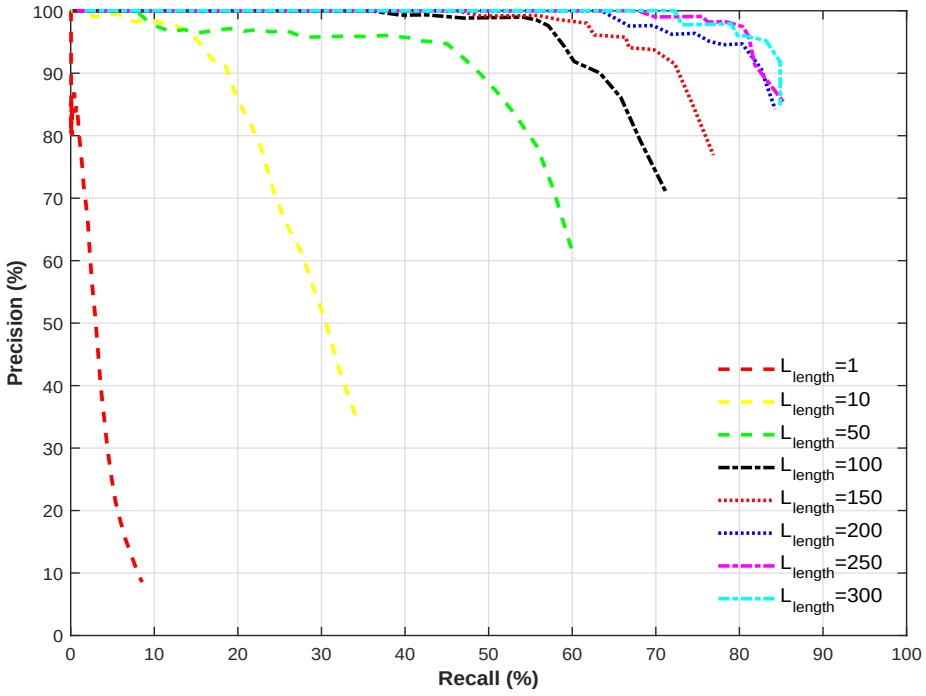
Figure 4-6: Performance of the proposed method according to different used features and comparison with SeqSLAM method (summer vs winter, sequence length L_{length}=200).

100% precision, which outperforms a little LBP++GIST and significantly the SeqSLAM method. It should be noted that the image size used in SeqSLAM is also 32×32 and the other parameters of SeqSLAM method correspond to default situation as used in [100].

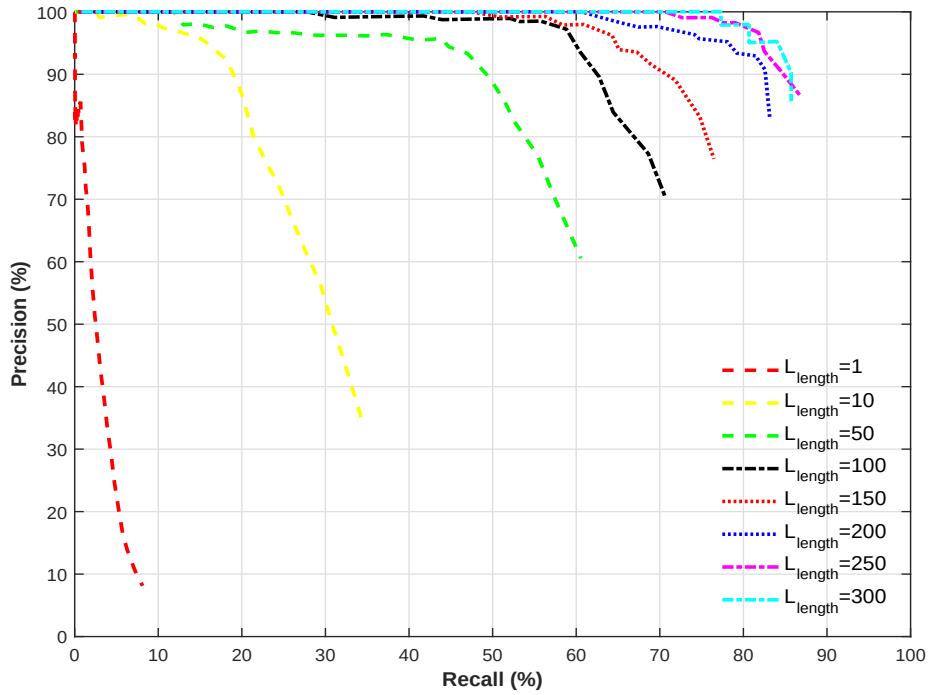
4.4.2 Sequence length selection

Traditionally visual localization has been performed by considering places represented by single images. Recently, several approaches such as SeqSLAM, have proved that recognizing places through sequences of images is more robust and effective [72]. In this chapter, we also follow the idea of using sequences of images instead of single image for identifying places. This approach allows to achieve better results for visual localization in different seasons, as it can be seen in Fig.4-7.

Fig.4-7 shows the performance achieved when varying sequence length between 1 and 300 frames for two different feature combination: LBP++GIST and CSLBP++GIST. Sig-



(a) LBP++GIST



(b) CSLBP++GIST

Figure 4-7: Performance comparison of our proposed method with different feature combination, according to image sequence length L_{length} (spring vs winter).

nificant performance improvement was achieved by increasing the sequence length up to 200 frames, after which the improvement became modest. According to the precision-recall curves demonstrated in Fig.4-7, the influence of sequence length (L_{length}) is decisive for improving the performance of visual localization in different seasons. Moreover, there is a limit near to a length of 200 frames, from which the results are not greatly enhanced. For this reason, sequence length L_{length} is set to 200 frames in the rest of the experiments.

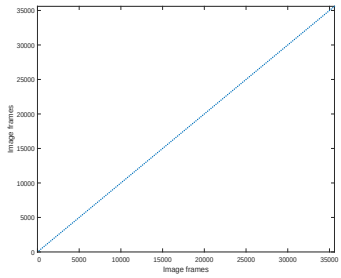
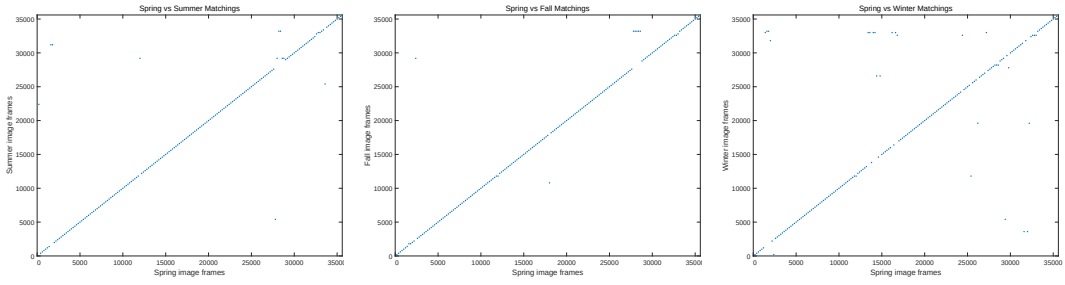


Figure 4-8: Ground-truth. Since frame from one season corresponds to the same frame in any of the other three seasons, the ground-truth is diagonal.



(a) spring vs summer (b) spring vs fall (c) spring vs winter

(d) summer vs fall (e) summer vs winter (f) fall vs winter

Figure 4-9: Matching results under different season couples at 100% recall situation. The expected result is along the diagonal.

4.4.3 Visual localization under different season couples

After feature performance evaluation and sequence length selection, visual localization using sequence matching based on multi-feature combination (CSLBP++GIST) was compared under different season couples.

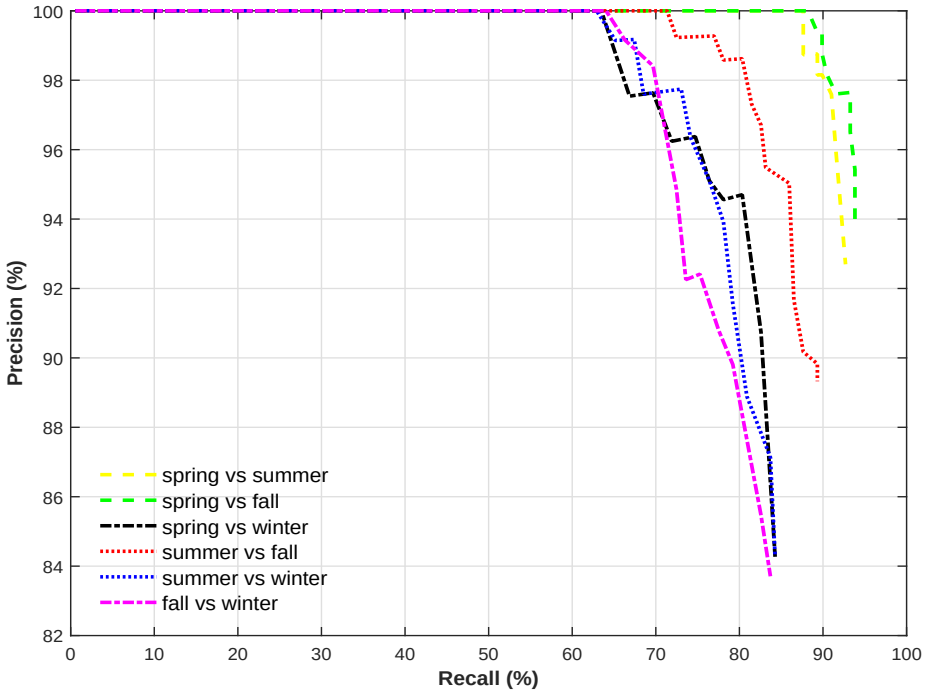
Fig.4-8 illustrates the ground-truth of image matching (for every possible couple of seasons). It should be noted that the position of the train in an arbitrary frame in one season corresponds to the same frame in any of the other three seasons thanks to time-synchronization.

The matching results under different season couples are depicted in Fig.4-9. As our objective is to correctly identify the place as much as possible (along the diagonal), it can be seen that the result of "summer vs fall" is the best among the others. It can be also noticed that when the winter sequence is evaluated (Fig.4-9 (c), (e) and (f)), the number of unrecognized places increase, that is because the snow in winter leads to featureless scenes.

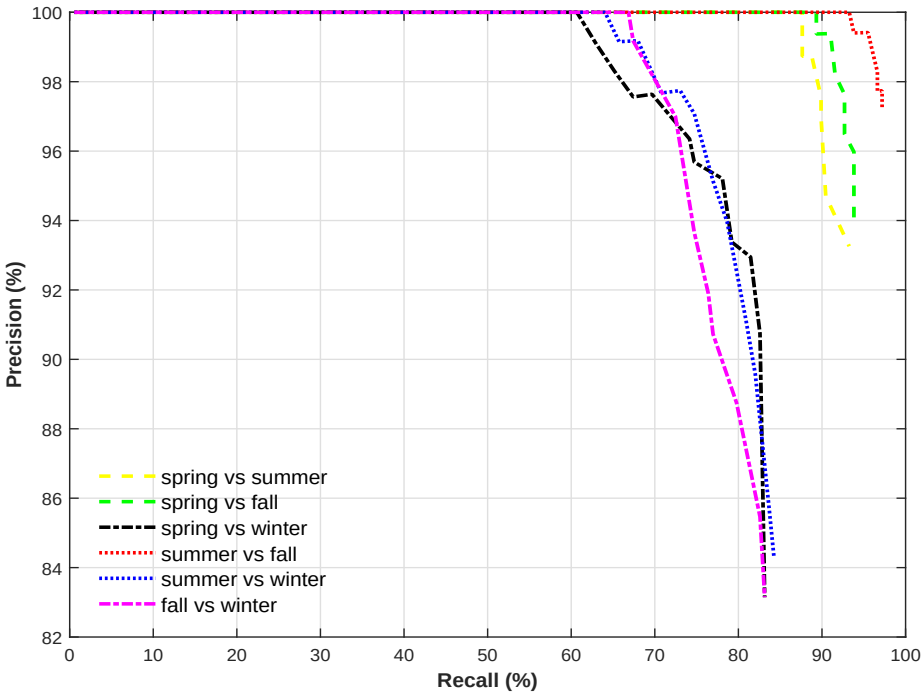
Fig.4-10 shows precision-recall curves of matching results under different season couples. It can be easily found that visual recognition performance of our method is better under (spring vs summer), (spring vs fall) and (summer vs fall), where we can reach above 85% of recall at 100% precision level. It can be seen also that our proposed multi-feature combination method can achieve recall rate above 60% at 100% precision under all the season couples. The overall performance of CSLBP++GIST is better than that of LBP++GIST. As expected, when winter sequence is evaluated, the effectiveness of our method decreases due to the extreme changes that this season causes in place appearance because of environmental conditions such as presence of snow, illumination and vegetation changes, etc.

Fig.4-11 shows an example of frame matches using the proposed method. Despite the large variations in appearance (many vegetations in fall while snow covering the ground in winter), place recognition using the multi-feature attained good matching performance.

For the visual localization based on place recognition, we are primarily interested in the recognition rate high precision level. The recall scores for high selected precision values of SeqSLAM method and our proposed approach are given in Table 4.1. For all the cases, our



(a) LBP++GIST



(b) CSLBP++GIST

Figure 4-10: Precision-recall curves comparing the performance of different feature combination under different season couples.

(a) Fall (b) Winter

Figure 4-11: Corresponding frames from sequence matching between two seasons (fall and winter). The left column shows fall image frames queried from winter traversal, and the right column shows the winter image frames recalled by our approach.

proposed place recognition based visual localization algorithm achieves the better recall rate. Moreover, in " spring vs summer " and " spring vs fall " situations, the recall rates of our approach is higher than 85% for all the high precision values recorded in Table 4.1.

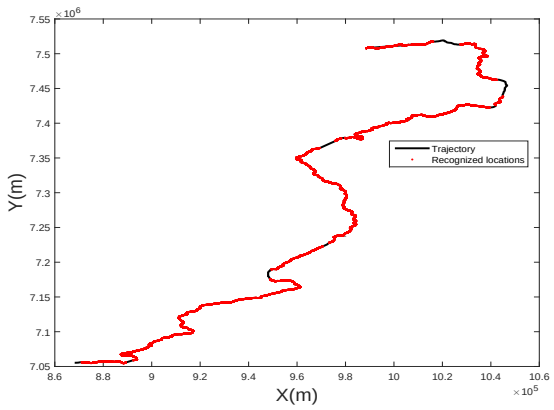
Table 4.1: Recall scores at selected high precision levels (100%, 99%, 90%)

Different season couples	Method	100% precision	99% precision	90% precision
Spring vs Summer	SeqSLAM	20.45	27.73	66.11
	LBP++GIST	87.64	87.64	92.70
	CSLBP++GIST	87.64	87.64	93.26
Spring vs Fall	SeqSLAM	15.41	27.45	63.87
	LBP++GIST	88.20	89.89	93.82
	CSLBP++GIST	89.33	91.01	93.28
Spring vs Winter	SeqSLAM	14.29	17.37	62.18
	LBP++GIST	63.48	66.58	82.58
	CSLBP++GIST	60.67	62.92	82.58
Summer vs Fall	SeqSLAM	9.80	23.81	65.27
	LBP++GIST	71.35	76.97	87.64
	CSLBP++GIST	93.26	95.51	97.19
Summer vs Winter	SeqSLAM	14.01	27.45	53.50
	LBP++GIST	62.92	67.42	79.21
	CSLBP++GIST	64.04	67.98	80.90
Fall vs Winter	SeqSLAM	2.24	2.35	44.82
	LBP++GIST	64.40	66.29	77.53
	CSLBP++GIST	66.85	67.42	76.97

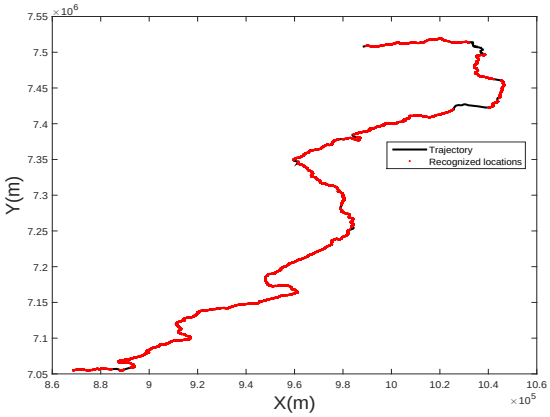
For both SeqSLAM method and our approach, recall rate increases when precision is decreasing. The recall rate of the two methods increases drastically at 90% precision. Besides that, the recall rate of SeqSLAM method is lower than the recall rate of the proposed method, and worst for all the high precision values. This is probably due to the fact that the SeqSLAM method has a certain dependence of the field of view and the image size, as demonstrated in [100].

Fig.4-12 shows visual localization results of different season couples at 100% precision level. It can be seen that most places can be successfully localized, and at least 60% of the places (red points) can be localized in the worse matching case (spring vs winter).

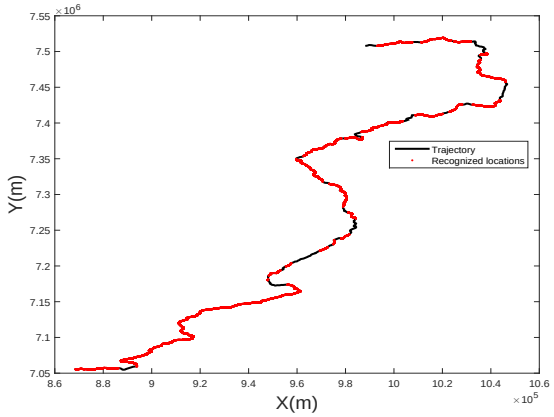
In addition, the computational time of sequence matching using the combined feature (CSLBP++GIST), is illustrated in Table 4.2. The computational time is increasing when



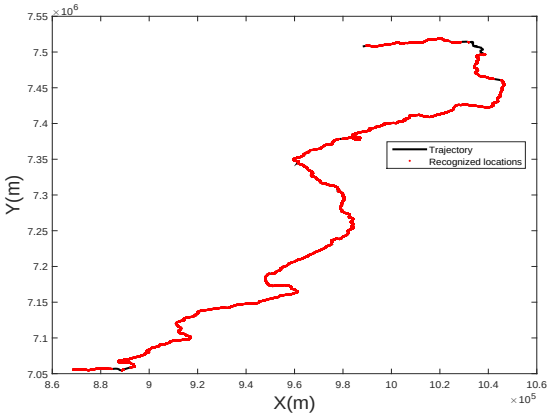
(a) spring vs summer



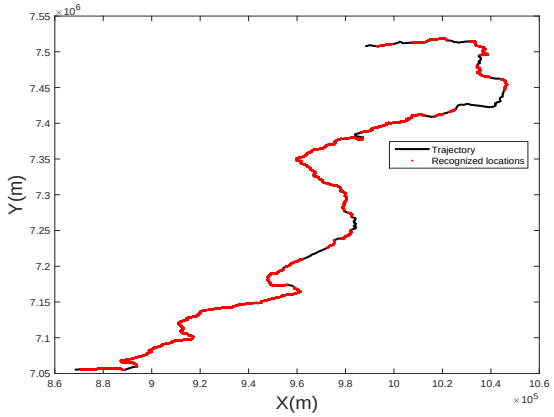
(b) spring vs fall



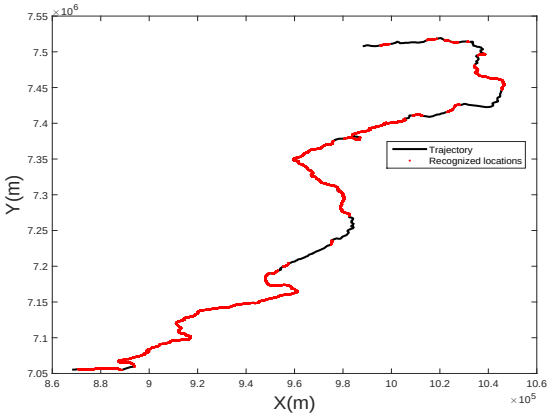
(c) spring vs winter



(d) summer vs fall



(e) summer vs winter



(f) fall vs winter

Figure 4-12: Visual localization results under different season couples at 100% precision level. Successful matched images that come from the same location (on the basis of appearance alone) are marked in red points.

the size of training database is large. Since CSLBP++GIST feature dimension is lower that of LBP++GIST, so the processing time of CSLBP++GIST based matching is shorter. Compared with SeqSLAM, the proposed method using CSLBP++GIST feature is faster. The experiment were conducted on a Intel Core i7, 2.40 GHz laptop.

Table 4.2: Comparison of average processing times for image matching(/s)

No. images in database	SeqSLAM	LBP++GIST	CSLBP++GIST
200	3.5335	3.1280	2.9463
2000	54.6982	20.4664	18.3309
20000	87.6982	33.4664	37.3309

4.5 Conclusion and Future Works

In this chapter, we proposed a feature combination based sequence matching method to perform robust localization even under substantial seasonal changes. After feature extraction, Chi-square distance is used to measure similarity between a testing sequence and the training sequences of a training database. A distance ratio is then calculated before applying a thresholding procedure to validate the good matching candidates.

Thanks to precision-recall based evaluation, experimental results showed that the proposed sequence matching method is more robust and effective for long-term visual localization in challenging environments. The proposed method takes advantages of local and global image information, which can reduce aliasing problem. Sequence length analysis demonstrated that sequences as long as 200 frames could provide viable recognition results. Shorter sequences cannot achieve acceptable results, while longer ones cannot bring significant improvement. Compared to the state of the art SeqSLAM method, the proposed approach provides better recognition performances. In addition, according to the localization results, at least 60% of the places can be localized using the appearance through the proposed method.

However, using feature combination increases feature vector dimension and thus increases time computation. To overcome this limitation, we envision to deal with dimension

reduction using space projection techniques or searching methods like local sensitive hashing (as done in the approach presented in Chapter 3). Another drawback is that, in the performed experiments, testing and training sequence lengths are same as that twice driving speeds of train are very close. In the future, more flexible sequence length selection and matching strategy should be considered.

Chapter 5

All-environment Visual Localization Based on ConvNet Features and Localized Sequence Matching

In Chapter 4, the used hand-craft feature CSLBP++GIST and sequence matching technique bring some benefits for visual localization in some simple driving traffic environment. In this chapter, for further study, deep learning features (ConvNet features) instead of hand-craft features are used to strengthen place describing ability, and localized sequence matching is developed to improve sequence matching performance. Therefore, the advantages of ConvNet features obtained from convolutional neural networks and localized sequence matching technique are investigated in this chapter.

5.1 Introduction

As illustrated in the previous chapters, place recognition based visual localization can be achieved by sophisticated hand-crafted features. However, as robots or vehicles operate for longer periods of time in real-world environments, the huge variations on the visual perception of a place caused by factors such as different days, varying weather conditions and seasonal changes, remain a significant challenge for place recognition based visual localization. Though many advances have been made in the recent past [72, 104], improv-

ing place recognition accuracy and reliability for visual localization, still remains an open problem.

There are particular issues in the domain of environments and appearance changes (such as illumination changes, across seasons, structural changes in the environment, etc.) that affect place recognition accuracy, which need to be addressed to achieve all-environment visual localization. Therefore, there is a need to have robust place recognition systems which have strong place describing and recognizing abilities that can deal with these changes (appearance and illumination invariants). The biggest advantage of a such system would be for all-environment localization across months or years, as there will not be a need to update the map with multiple copies of the same location under different conditions.

Currently, deep learning applied in computer vision areas can help to robustly solve the previously mentioned dilemmas associated with all-environment visual localization. Deep learning methods aim at learning feature hierarchies with features from higher levels of the hierarchy formed by the composition of lower level features. Automatically learning features at multiple levels of abstraction allow a system to learn complex functions mapping the input to the output directly from image data, without depending completely on human-crafted features [11]. Convolutional neural network (ConvNet) as one of promising deep learning which removes complicated and problematic hand-crafted feature engineering, are sensitive to small sub-regions of visual field which are well-suited to exploit the strong spatially local correlation present in natural images [66]. ConvNet features extracted from convolutional neural network have been demonstrated to be versatile and transferable that is, even though they have be trained to solve a particular task, they can be used to solve different problems [101].

The supervised deep convolutional neural networks permit to deliver high level performance on most of challenging classification tasks [41]. Through training on large amounts of labeled data, millions of network parameters can be optimized which makes the deep networks robust and powerful. Once trained, ConvNets obtain discriminative and human-interpretable feature representations [69]. Therefore, it is practicable to develop powerful and robust visual localization systems by taking advantage of ConvNet features.

In this chapter, the problem of all-environment visual localization is addressed by de-

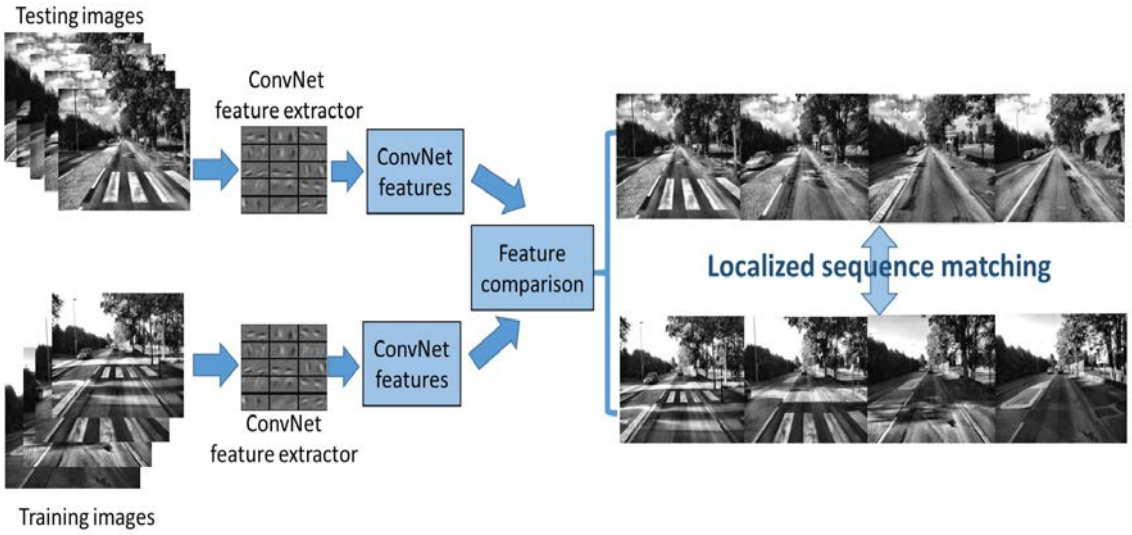


Figure 5-1: Schematic illustration of visual localization using ConvNet features and localized sequence matching. ConvNet features are extracted from testing images and then compared to those extracted from all images of the training database. After feature comparison, localized sequence matching is conducted to find the best image matching.

veloping a localized sequences matching based place recognition framework using ConvNet features. The visual localization framework centered around ConvNet features and localized sequence matching is illustrated in Fig.5-1. ConvNet features are firstly extracted using a pre-trained network, then the features are compared using cosine distance. Finally, localized sequence matching is conducted to recognize the current place.

In our work, we exploit the hierarchical nature of ConvNet features and compare different ConvNet layers for place recognition under severe appearance and illumination variations. Furthermore, a comparison with state-of-the-art place recognition methods is performed on four datasets. The F_1 scores attained with the conv4 layer of ConvNet for the four different datasets are higher than 0.85, which is significant better than those of FAB-MAP and SeqSLAM. At last, for real-time visual localization consideration, a speed-up method is achieved by approximating the cosine distance between features with hamming distance over bit vectors obtained by Locality Sensitive Hashing (LSH). Using 4096 hash bits instead of the original feature permits to accelerate by 12 times the computation time, retaining 95% of original place recognition performance.

The chapter structure is as follows. In Section 5.2, the components of the proposed

place recognition based visual localization system are described. Experimental setup is described in Section 5.3, and results are presented in Section 5.4. Finally, the chapter is concluded and future works are discussed in Section 5.5.

5.2 Proposed Approach

The proposed visual localization approach can be divided into off-line and on-line parts. In the off-line part, a set of GPS tagged training images $I^{train} = \{I_i^{train}\}_{i=1}^{N^{train}}$ is firstly acquired, where N^{train} is the number of training images. Then, the pre-trained caffe-alex network (trained using ILSVRC2012 dataset) is used to extract features from training images [54]. The extracted ConvNet features from training database are noted $F^{train} = \{f_i^{train}\}_{i=1}^{N^{train}}$, where f_i^{train} is the feature extracted from the training image I_i^{train}. For the on-line phase, the current testing image I_T^{test} is input into caffe-alex network and the ConvNet feature f_T^{test} of the current testing image is computed. Then, f_T^{test} is compared with the training image feature set $\{f_i^{train}\}_{i=1}^{N^{train}}$ using cosine distance (Section 5.2.2).

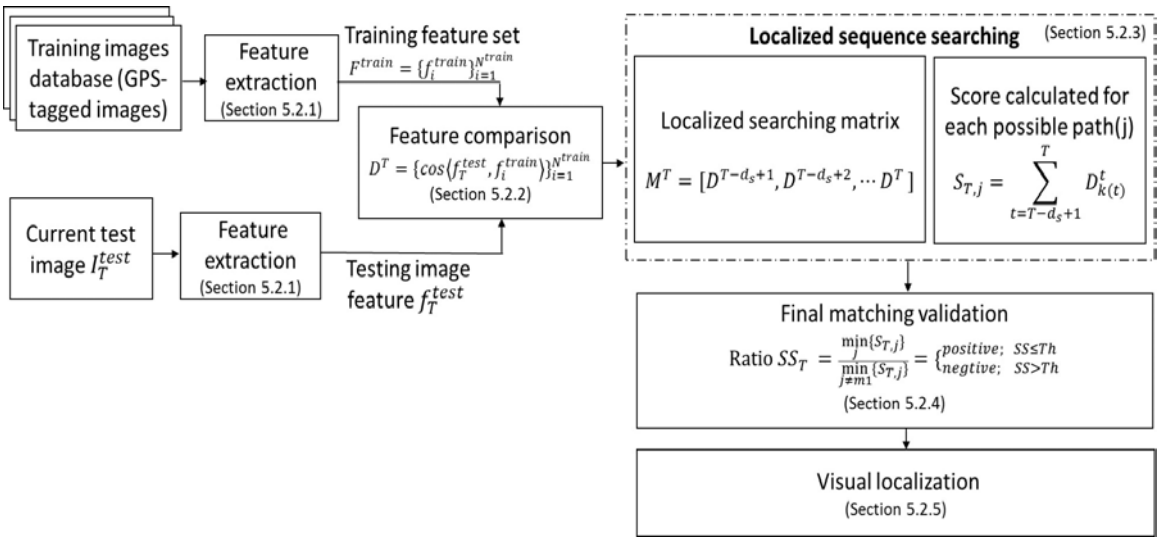


Figure 5-2: Detailed block diagram of the proposed visual localization method. Feature extraction uses pre-trained network, feature comparison uses cosine distance and localized sequence searching is based on potential paths.

In terms of localized sequence matching, given a testing sequence of length d_s (the sequence is composed of images indexed from $T - d_s + 1$ to T, where T is the index of

the current image), some possible training sequence candidates are firstly determined from the training database through the ratio between the testing and training trajectory speeds. For each possible training sequence candidate, a score S is calculated by summing all the cosine distances along the sequence (Section 5.2.3). The sequence candidate that provides the minimum score can be considered as the most similar one to the testing sequence. In fact, the two best sequences (according to matching score) are conserved for further validating the final matching result.

Following, the best matching candidate will be validated through a distance ratio SS (see Section 5.2.4). This distance ratio SS of the two minimum computed distances (corresponding to the two best candidates) is considered to validate the training sequence that finally best matches to the current testing sequence. If the ratio SS is below or equal to a threshold Th, the first best sequence candidate (with the lower matching score) is confirmed and regarded as positive, otherwise it is considered as negative one (in this case, no matching result is conserved). When a sequence candidate is confirmed as positive, the position can be obtained from the matched GPS-tagged training images (see Section 5.2.5).

As illustrated in Fig.5-2, there are four important components in our visual localization approach:

- **ConvNet features extraction** (detailed in Section 5.2.1): ConvNet features F^{train} are extracted from all training database images by off-line processing and f_T^{test} is extracted from current testing image by on-line processing, using the pre-trained caffe-alex network. These learned features are robust to both appearance and illumination changes and allow representing each location (place) very well. The extracted ConvNet features will be compared in the next step.

- **Feature comparison** (detailed in Section 5.2.2): The cosine distances are computed between the feature f_T^{test} of the current testing image and the features $\{f_i^{train}\}_{i=1}^{N^{train}}$ of all the images of the training database. All these distances formed a vector D^T. Based on this, localized sequence matching will be conducted in the next step.

- **Localized sequence matching** (detailed in Section 5.2.3): To achieve efficient place recognition, localized sequence matching is used instead of single image match-

ing. Considering the testing sequence composed of the last d_s testing images (indexed from $T - d_s + 1$ to T), sequence matching is conducted in the matrix $M^T = [D^{T-d_s+1}, D^{T-d_s+2}, \cdots, D^T]$. According to the ratio between the testing and training trajectory speeds, some possible training sequence candidates in the training database can be firstly determined. A score S is calculated by summing all the testing image to training image cosine distances along each possible sequence candidate. The sequence that provides the minimum score can be considered as the most similar one to the testing sequence. The two best sequence matching scores are conserved for further matching validation.

- **Final Matching Validation** (detailed in Section 5.2.4): For each testing sequence, the best training sequence candidate will be validate in this step to reduce some false recognition. The ratio SS between the two best sequence matching scores is used to verify the best sequence candidate. If the ratio SS is below or equal to a threshold Th, the first candidate (with the lower matching score) is confirmed and regarded as positive matching, otherwise it is considered as negative one (in this case, no matching is conserved).

Several advantages of our approach can be highlighted:

1) The system uses an off-the-shelf pre-trained convolutional network to extract features which makes feature extraction more conveniently.

2) ConvNet features as auto-learned features are more stable and powerful. By using these robust features as descriptors for place representation, we inherit their robustness against appearance and weather changing.

3) Using a localized sequence matching allows us to search in a small range rather than in the whole training database. This makes place recognition more robust and efficient.

5.2.1 ConvNet features extraction

In this work, caffe-alex [45] ConvNet pre-trained model (provided by MatConvNet) and MatConvNet toolbox [105] are deployed to extract features. The caffe-alex ConvNet model is a 21 layers network (see Fig.5-3) which is mainly constructed by five different layer

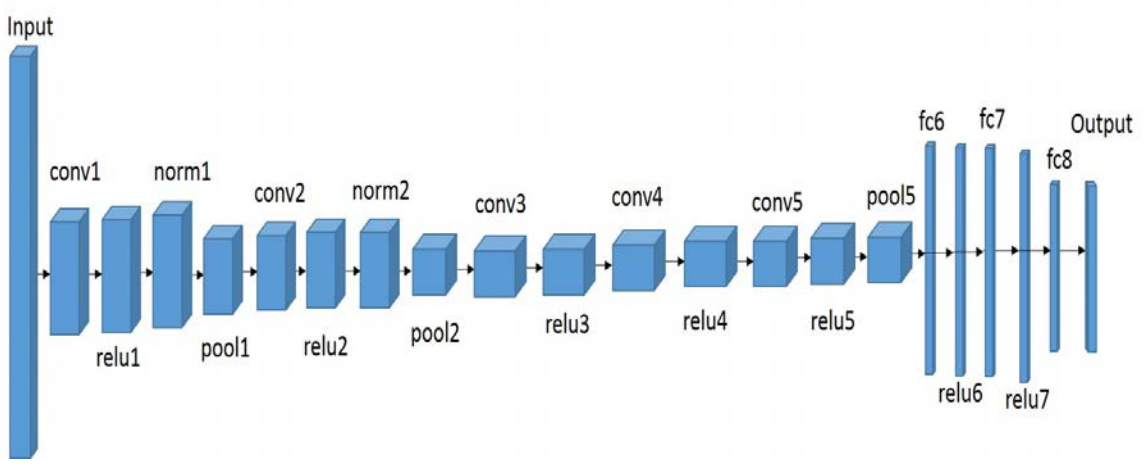


Figure 5-3: Architecture of the caffe-alex network. ConvNet transforms the original image, layer by layer, from the original pixel values to the final class scores.

types: convolutional layer (conv), pooling layer (pool), rectified linear units layer (relu), normalization layer (norm) and fully-connected layer (fc).

Among these layers, conv layers compute the output of neurons which are the core building blocks of a convolutional Network. The output of conv layers can be interpreted as holding neurons arranged in a 3D volume. Relu layers apply an element-wise activation function. Pool layers perform a down-sampling operation along the spatial dimensions. Norm layers can be used to get some kind of inhibition scheme. The fc layers, as fully-connected layers, compute the class scores.

Let consider an image x and ConvNet learned parameter w. From the input layer (taking the image x) of the network, a sequence of layered outputs is produced. Each layer output is the input of the next layer. Thus, the original input image is transformed layer by layer to the final class scores:

$$f(x) = f_{21}(\dots f_2(f_1(x;w_1);w_2)\dots),w_{21}) \tag{5.1}$$

where $f_1, \cdots, f_{21}$ are the corresponding layer functions as illustrated in Fig.5-3. Each layer output is a deep learnt representation of the image (ConvNet feature). The low layers retain high spatial resolution with low-level visual information. While high layers capture more semantic information and less fine-grained spatial details. The network is able to process images of any size equal to or greater than 227×227 pixels (the original caffe-alex net-

work was trained on 227×227 images). Place recognition is then performed by comparing the ConvNet features extracted from current testing image I_T^{test} with the ConvNet features extracted from all the images $\{I_T^{train}\}_{i=1}^{N^{train}}$ of the training database.

Considering that middle layers take the advantage of both low-level and semantic information, our approach exploits feature information of these middle layers to handle large appearance changes and then alleviate false recognition. The used layers and their dimensionality are listed in Table 5.1. The corresponding ConvNet features generated by convolutional Networks from an example of input image are illustrated in Fig.5-4. It can be seen that conv4, conv5 and relu5 layers provide more image spatial information while pool5, fc6 and fc7 layers bring more semantic information.

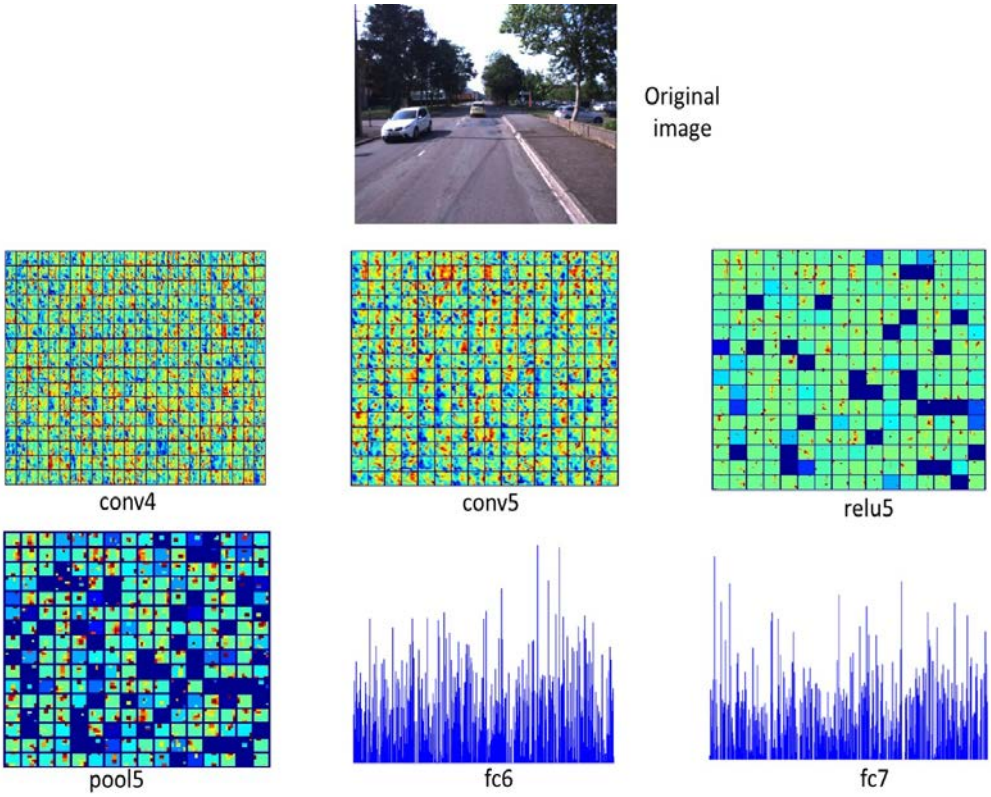


Figure 5-4: An example of a scene and extracted features from different layers of the caffe-alex network. Features obtained from different ConvNet layers can serve as holistic image descriptors for place describing.

Table 5.1: The layers from the caffe-alex ConvNet model used in our evaluation and their output dimensionality (height×width×feature map number).

Layer	Dimensions	Layer	Dimensions
conv4	$13 \times 13 \times 384$	pool5	$6 \times 6 \times 256$
conv5	$13 \times 13 \times 256$	fc6	$1 \times 1 \times 4096$
relu5	$13 \times 13 \times 256$	fc7	$1 \times 1 \times 4096$

5.2.2 Feature comparison

Feature comparison is performed based on the cosine distance between ConvNet features. Each test image feature is compared with the image features of all the training images. For that, the cosine distances between the feature f_T^{test} of the current testing image I_T^{test} and the features $\{f_i^{train}\}_{i=1}^{N^{train}}$ of all the images of the training database are computed as follows :

$$d_{T,i} = cos\langle f_T^{test}, f_i^{train} \rangle = \frac{f_T^{test} \cdot f_i^{train}}{\|f_T^{test}\| \|f_i^{train}\|};\ i = 1, 2, \cdots, N^{train} \tag{5.2}$$

Then, these N^{train} distances are concatenated to form a vector $D^T \in \mathbb{R}^{N^{train} \times 1}$:

$$D^T = [cos\langle f_T^{test}, f_1^{train} \rangle, cos\langle f_T^{test}, f_2^{train} \rangle, \cdots, cos\langle f_T^{test}, f_{N^{train}}^{train} \rangle] \tag{5.3}$$

where N^{train} is the total number of images in training database. D^T is the vector that contains the cosine distance between the testing image I_T^{test} and all the training images. It is represented as a column in a matrix as shown in Fig.5-5.

5.2.3 Localized sequence matching

Assume that the vehicle travels in repeated route with negligible relative acceleration. For a given testing sequence, composed of d_s images, indexed from $T - d_s + 1$ to T, where T is the index of the current testing image, we search the corresponding sequence (from the training database) in a local matrix rather than in the whole training database. This searching procedure is performed by considering possible training sequence candidates, that are determined by the speed ratio between the training and testing trajectories. This

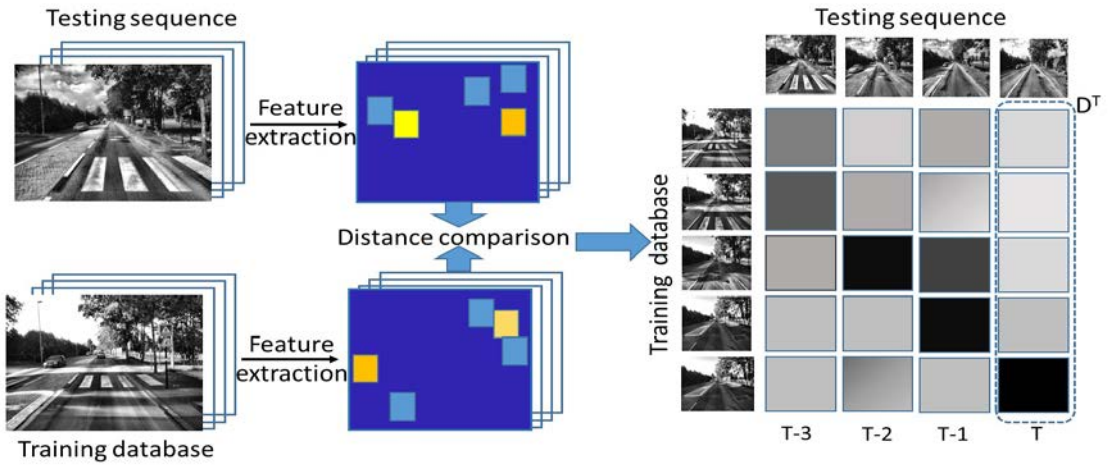


Figure 5-5: Procedure for ConvNet features comparison. Features are extracted from each testing image and then compared with features extracted from all training images.

procedure is qualified as localized sequence searching.

As Fig.5-5 shows, for each new testing image I_T^{test}, localized sequence searching is performed through a matrix M^T constructed by cosine distance vectors $D^t(T-d_s+1 \leq t \leq T)$ over the test sequence, composed of the d_s previous images (including the current testing image):

$$M^T = [D^{T-d_s+1}, D^{T-d_s+2}, \cdots, D^T] \tag{5.4}$$

where d_s is the testing sequence length (in terms of images number) that determines how far back the search goes. As defined previously, $D^t(T-d_s+1 \leq t \leq T)$ is the cosine distance column vector for the testing image I_t^{test}. It contains the distances between the testing image feature f_t^{test} and all training image features $\{f_i^{train}\}_{i=1}^{N^{train}}$.

Due to the linear relationship between the speed of training and testing trajectories, the possible paths representing different speed ratio can be projected into each element in the matrix M^T. Thus, the lowest-cost path, which has the minimum distance score S, is deemed to be best match as the red line shown in Fig.5-6.

As shown in Fig.5-6, each element of the matrix M^T is the cosine distance between a testing image and a training image. The blue color in the matrix M^T indicates small distance value while the red color means large distance value. Searching range are constrained into the space between minimum speed V_{min} and maximum speed V_{max}. Each possible path

(dark line) in the space indicates a possible match between testing (query) sequence and training sequence. The lowest-cost path (red line) is regarded as the best matching.

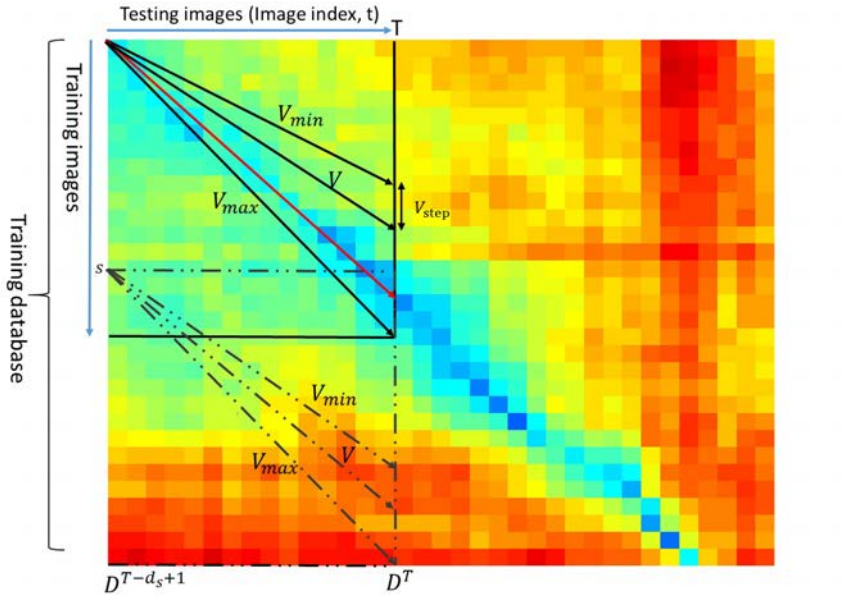


Figure 5-6: The search algorithm finds the lowest-cost straight line within the searching matrix M^T. These lines are the set of potential paths through the matrix. The red line is the lowest-cost path which aligns the testing sequence and training sequence. Each element indicates the cosine distance between two images (test and train).

A difference score S is calculated for each path based on the distance values the line passes from frame number $T-d_s+1$ to the current frame T:

$$S = \sum_{t=T-d_s+1}^{T} D^t_{k(t)} \tag{5.5}$$

where $k(t)$ is the index of the column vector D^t by which the path (line) passes through:

$$k(t) = s + V(t-(T-d_s+1)) \tag{5.6}$$

where s is the training image index from which the path is originated. The initial value of s is 0, then increased by 1 at each step. V is the vehicle speed varying between V_{min} and V_{max} with a step value V_{step}. The score S (sum of distance values along path (line)) is used to identify the best matching candidate (who has the lowest score) for each testing sequence.

5.2.4 Final matching validation

Given the current testing image number T, the corresponding testing sequence T (images indexed from $T-d_s+1$ to T) can be constructed. Using the localized sequence matching method, the best two sequence candidates who have smaller scores are conserved for further validation. Suppose S_{m1} and S_{m2} are respectively the first and second minimum score of the top two training sequence candidates to the testing sequence, where

$$\begin{cases} S_{T,m1} = \min\limits_{j}\{S_{T,j}\} \\ S_{T,m2} = \min\limits_{j(j\neq m1)}\{S_{T,j}\} \end{cases} \tag{5.7}$$

In order to verify the best sequence matching, a ratio SS_T is calculated as follows:

$$SS_T = \frac{S_{T,m1}}{S_{T,m2}} \tag{5.8}$$

The value of ratio SS is between 0 and 1. A threshold Th is then applied to the ratio SS_T to determine if the sequence pair $(T, m1)$ is matched or not. If the ratio SS_T is not larger than the threshold Th, which means the training sequence corresponding to $m1$ is matched to the current testing sequence, this is also called positive matching. Otherwise no matching is considered (negative matching).

5.2.5 Visual localization

After a matching result is successfully validated, the vehicle can localize itself through the matched training image position. Since the training images are tagged with the GPS information, the vehicle can get its position information through the training image matched with the current testing image. As for the approaches presented in the two previous chapters, this is also a topological level localization—simply identifies the most likely location.

5.2.6 Algorithm of proposed visual localization

Algorithm 5.1 illustrates the ConvNet features and localized sequence matching based visual localization.

Algorithm 5.1 Localized sequence matching based visual localization.

Inputs:

$\{I_i^{train}\}_{i=1}^{N^{train}}$ {training images database}; $\{I_i^{test}\}_{i=1}^{N^{test}}$ {testing images database};
N^{train}, N^{test} {training and testing images numbers};
V_{max}, V_{min} {maximum and minimum vehicle speeds}; V_{step} {Vehicle speed step-size}
d_s {Sequence length} ;

Outputs:

S {Path-line (sequence candidate) score };
Vehicle position;

Algorithm:

/* ConvNet features extraction; {section 5.2.1} */
for $i \leftarrow 1$ to N^{train} **do**
  $F^{train} = \{f_i^{train}\} \leftarrow$ Feature extraction for training images;
end for
for $i \leftarrow 1$ to N^{test} **do**
  $F^{test} = \{f_i^{test}\} \leftarrow$ Feature extraction for testing images;
end for

/* Feature comparison */
for $i \leftarrow 1$ to N^{test} **do**
  for $j \leftarrow 1$ to N^{train} **do**
    $d_{i,j} \leftarrow \cos\langle f_i^{test}, f_j^{train} \rangle$; // Computation of the cosine distance between the test image I_i^{test} and the training image I_i^{train} {Section 5.2.2}.
  end for
  $D^i \leftarrow [d_{i,1}, d_{i,2}, \cdots, d_{i,N^{train}}]$; Column vector $D^i \in \mathbb{R}^{N^{train} \times 1}$ that contains the cosine distance between the testing image I_i^{test} and all the training images.
end for

/* Localized sequence matching and validation */
for $T \leftarrow d_s$ to N^{test} **do**
  $M^T \leftarrow [D^{T-d_s+1}, D^{T-d_s+2}, \cdots, D^T]$; // Construction of the local searching matrix.
  $j \leftarrow 1$; // Initialization;
  for $s \leftarrow 0$ to $(N^{train} - V_{max} \times d_s)$ **do**
    for $V \leftarrow V_{min}$:V_{step}: V_{max} **do**
      $S_{T,j} \leftarrow 0$;
      for $t \leftarrow (T - d_s + 1)$ to T **do**
        $k(t) \leftarrow s + V(t - (T - d_s + 1))$; // k is a line index in the column vector D^t; s is the training image number from which the path originated in (Section 5.2.3).
        $S_{T,j} \leftarrow S_{T,j} + D^t_{k(t)}$;// Score S is calculated for each possible path.
      end for
      $j \leftarrow j + 1$; Path-line number (sequence candidates) update;
    end for
  end for
  $SS_T = \frac{\min_j \{S_{T,j}\}}{\min_{j(j \neq m1)} \{S_{T,j}\}}$; $m1$ is the index of minimum score.
  if $SS_T <= Th$
    Vehicle position $\leftarrow$ The matched training images position
  if $SS_T > Th$
    Vehicle position $\leftarrow$ NaN (no position results)
end for

5.3 Experimental Setup

5.3.1 Datasets and ground-truth

Four datasets with different characteristics (as described in Table 5.2) will be used to evaluate our method.

Table 5.2: Description of the main characteristics of the datasets employed in the experiments.

Dataset	Length	No.images	Description
UTBM-1	2×4.0 KM	training:848 testing:819	minor variations in appearance and illumination
UTBM-2	2×2.3 KM	training:540 testing:520	medium variations in appearance and illumination
Nordland	4×728 KM	4×3577	severe variations in appearance
City Center	2×2.0 KM	training:1352 testing:1122	medium variations in appearance and illumination

1) UTBM-1 dataset

In this dataset (already used in Chapter3), the experimental vehicle (See Fig.1-4 in Chapter 1) traversed about 4 km in a typical outdoor environment (the trajectory can be seen in Fig.3-9 (a)). Some representative examples of UTBM-1 dataset is shown in Fig.5-7. From this figure, the changing of shadow, vegetation and field of view between the testing and training images can be also seen. As previously presented (Section 3.4.1 in Chapter 3), the training and testing data were collected respectively in 2014/9/11 and 2014/9/5. Among all the acquired images (at about 16 Hz), only a subset of images is selected to perform matching between the training and testing datasets (848 images for training and 819 images for testing). The average interval distance between two selected frames is around 3.5 m. Each image is associated with its GPS position obtained by a RTK-GPS receiver.

2) UTBM-2 dataset

The dataset UTBM-2 (also used in Chapter 3) consists of a 2.3 km long route in the urban city of Belfort, captured in 2014/9/5 (the trajectory can be seen in Fig.3-9 (b)). The

Figure 5-7: UTBM-1 dataset: Example of training and testing images (interval time of one week). Left column is training images and right column is testing images.

Figure 5-8: UTBM-2 dataset: morning vs afternoon. Left column is training images and right column is testing images.

first traversal of this dataset was performed in the morning and the second was conducted in the afternoon. The travel time of this dataset was approximately 20 minutes for the two traversals (training and testing). As shown in Fig.5-8, the illumination situation is different between the training and testing images. A total of 1060 images (540 and 520 images for the two traversals respectively) is used for the performance evaluation.

3) Nordland dataset

The Nordland dataset (already used in Chapter 4) consists of the video footage of a 728 km long train ride taken in northern Norway in four seasons [100]. As illustrated in Fig.4-5 of Chapter 4, there is a huge appearance variation between the four seasons. Due to seasonal changing, the different landscape (vegetation, mountains) and local weather conditions like sunshine, cloudy, rain and snowfall are experienced on the long trip. The original videos were recorded at 25 fps with a resolution of 1920×1080. The full-HD recordings have been time-synchronized such that the position of the train in an arbitrary frame from one video corresponds to the same frame in any of the other three videos. In our experiment, frames extracted from the original videos at 0.1 fps. GPS readings were recorded in conjunction with the video at 1 Hz.

4) City Center dataset

Figure 5-9: City Center dataset: twice traveling. Left column is training images and right column is testing images.

The City Center dataset was collected by Mobile Robotics Group of the University of Oxford [24]. The robot traveled twice around a loop with total path length of 2 km, and 2,474 images were collected by two (left and right) cameras mounted on the robot while traveling. This dataset was collected on a windy day with bright sunshine, which makes the abundant foliage and shadow features unstable, as can be observed in Fig.5-9.

For all the four datasets, ground-truth was constructed by manually finding pairs of frame correspondences based on GPS position.

5.3.2 Performance evaluation

As already described and used in the previous chapter, precision-recall curves and F_1 scores are still used in this chapter to evaluate the proposed approach. The final curve is computed by varying the threshold Th in a linear distribution between 0 and 1 and calculating the corresponding values of precision and recall. 100 threshold value are processed to obtain well-defined precision-recall curves.

In our experiments, the training images number is larger than or equal to the testing images number, thus each testing image has a ground-truth match. Therefore, among the positives, there are only true positives (correct results among successfully validated image matching candidates) and false positives (wrong results among successfully validated image matching candidates). The sum of the true positives and false positives is the total retrieved images number.

More specifically, precision is the ratio of true-positives over the retrieved images number (number of all the successfully validated images matching candidates), and recall is the ratio of true-positives over the total testing images. A perfect system would return a result where both precision and recall have a value of one. Based on the precision and recall, F_1 score can be defined as:

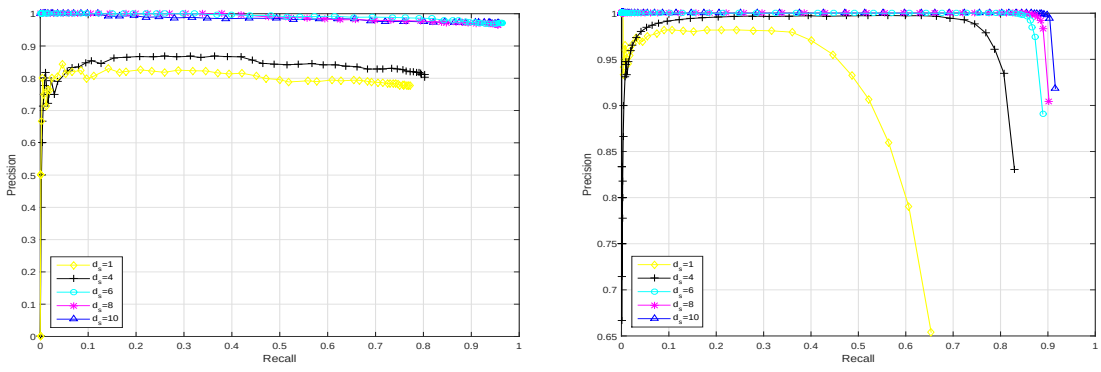
$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{5.9}$$

5.4 Experiments and Results

5.4.1 Performance comparison between single image and sequence based approach

Traditionally, visual localization has been performed by considering places as single images. However, other more recent proposals, such as SeqSLAM, changed this concept and introduced the idea of recognizing places as sequences of images.

In this section, the place recognition performances based on sequences of images and single images are compared. In Fig.5-10, results obtained for UTBM-1 and Nordland datasets are presented. Attending to the precision-recall curves depicted in Fig.5-10, the influence of the sequence length (d_s) is decisive to improve the performance of visual localization in life-long conditions. It can be clearly found that the approach using sequence allows to achieve better results than that of single image (almost no recall at 100% precision) in long-term visual localization.



(a) UTBM-1 (b) Nordland (spring vs summer)

Figure 5-10: Two examples of performance comparison of our proposal depending on the image sequence length (d_s) in the challenging UTBM-1 and Nordland (fall vs winter) datasets. The feature used here is conv4 layer.

Furthermore, there is a limit near to a length of 8 for UTBM-1 dataset and a length of 6 for Nordland dataset form which the results are not greatly enhanced. Based on this sequence length comparison and the driving speed, a sequence length of d_s=8 was chosen for datasets UTBM-1 and UTBM-2 in the rest of the experiments and results. For City Center dataset, sequence length was set to 3 and for Nordland dataset is 6. For all datasets,

Figure 5-11: Examples of frame matches from the Nordland dataset (fall vs winter). The top row shows testing frames, and the middle and third rows show the training frames recalled by $d_s = 1$ (single image) and $d_s = 6$, respectively. Visual recognition based on sequence matching achieves better performance than those obtained using single image.

velocity limits are V_{max} = 1.1 and V_{min}= 0.9, and a step size of V_{step} = 0.04 was set according to the experiment tests.

Fig.5-11 shows examples of frame matches on Nordland dataset (fall vs winter). Despite the large appearance variations between different seasons, the proposed ConvNet based visual localization using sequence matching (d_s=6) attained better recognition results than those obtained using single image (d_s=1).

5.4.2 ConvNet features layer-by-layer study

This section provides a thorough investigation of the utility of different layers in the ConvNet hierarchy for place recognition and evaluates their individual robustness against the two main challenges in visual place recognition: appearance and illumination changes.

Appearance change robustness

(1) UTBM-1 dataset: Fig.5-12 (top) illustrates images acquired in the training and testing datasets (interval time between the two acquisition is one week). The appearance has minor changes and the viewpoint has medium variations. The precision-recall curves are shown in Fig.5-12 (bottom). The recall obtained for the conv4 layer at totally correct level is around

40%. The performance of the layers fc6 and fc7 is poor.

UTBM-1 dataset: Example of training and testing images (interval time of one week)

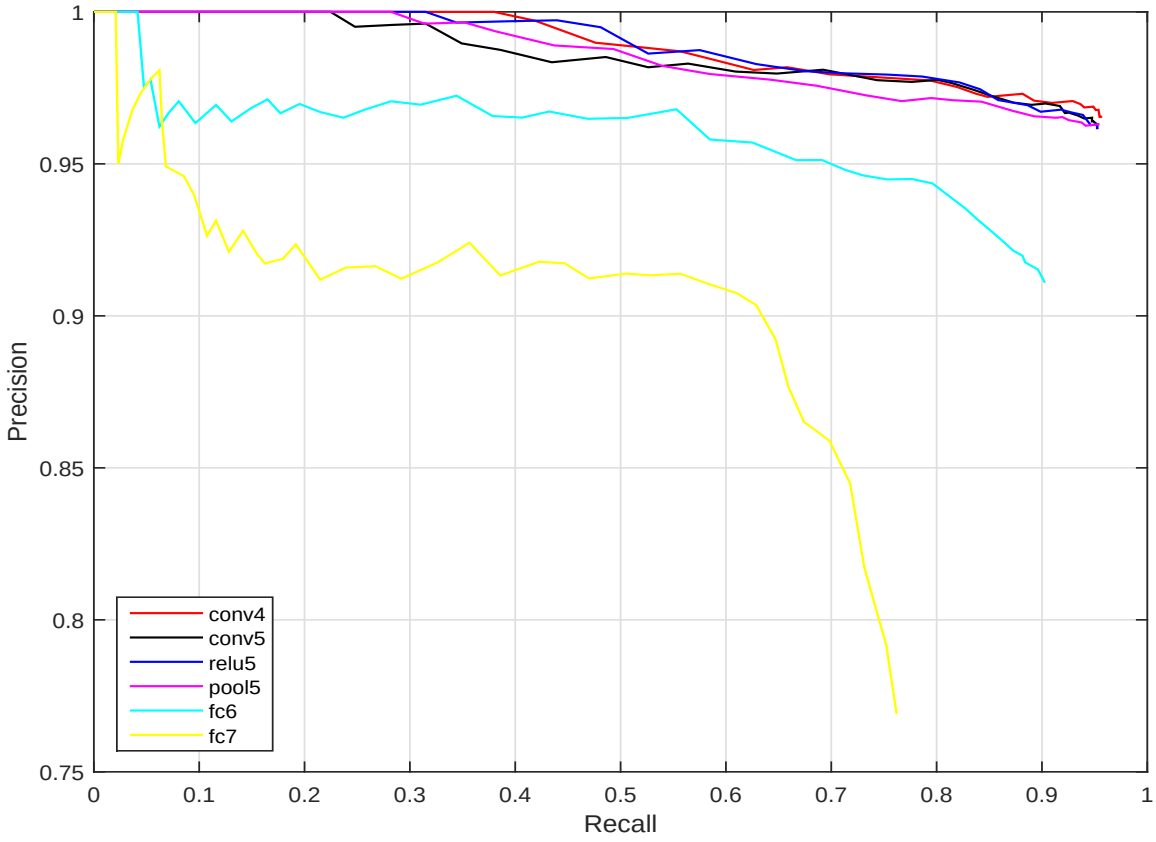


Figure 5-12: Precision-recall curves for UTBM-1 dataset (the trajectory acrosses forest, city and parking areas)($d_s = 8$).

(2) City Center dataset: This dataset was collected along public roads near the oxford city center with many dynamic objects such as traffic and pedestrians. In addition, it was collected on a windy day with bright sunshine, which makes the abundant foliage and shadow features unstable. The precision-recall curves are shown in Fig.5-13. Except that

the recall at 100% precision of the layer fc7 is less than 70%, the performance of the other layers (conv4, conv5, relu5, pool5 and fc6) reaches above than 75% recall at totally correct level. The conv4 layer is the best one achieving the highest recall level (above 80%).

City Center dataset: twice traveling

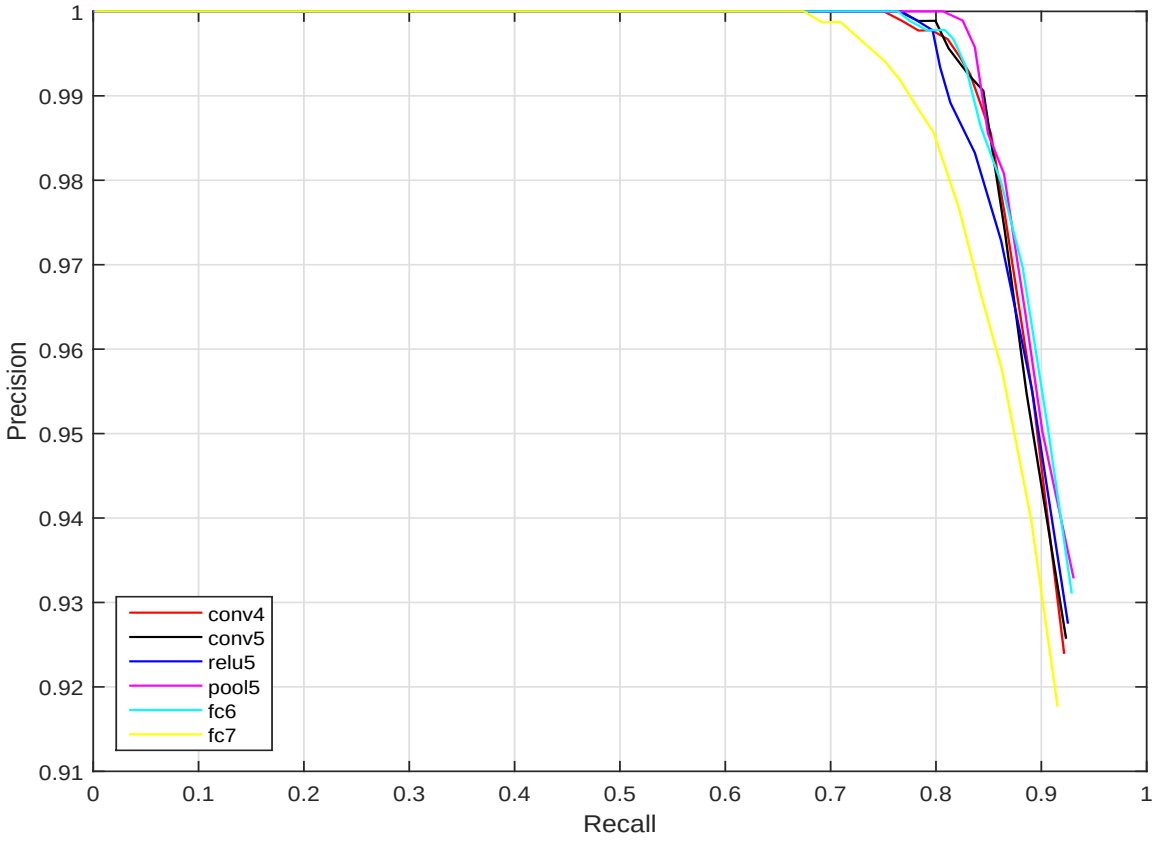
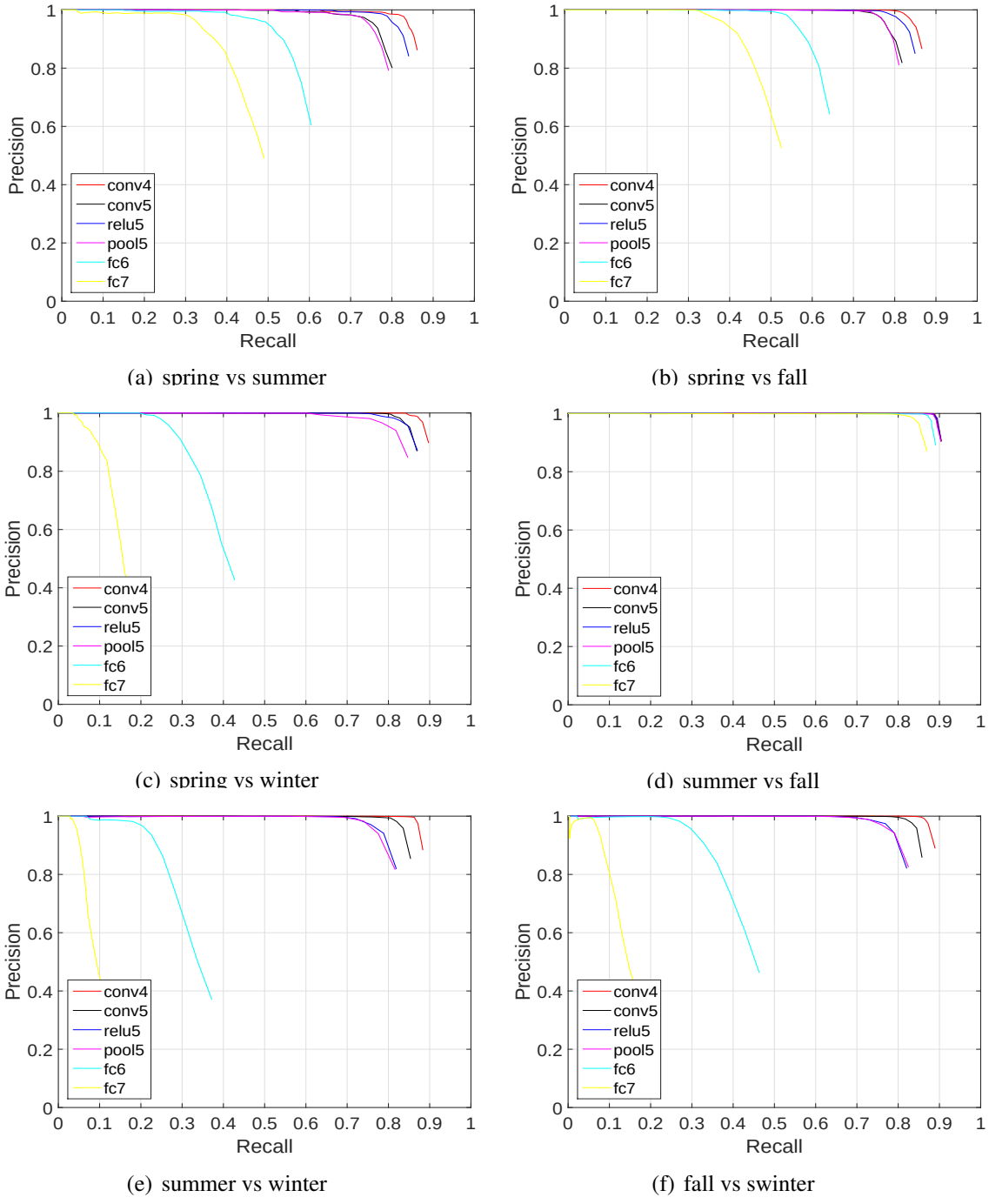


Figure 5-13: Precision-recall curves for City Center dataset (the trajectory acrosses city and parking areas)($d_s = 3$).

(3) Nordland dataset: It is probably the longest trajectory (3000 km) that can be currently used for life-long visual topological localization evaluation. It contains four videos

with very strong seasonal appearance changes. The precision-recall curves for different cases (one season for training vs another season for testing) are reported in Fig.5-14. It can



(a) spring vs summer (b) spring vs fall

(c) spring vs winter (d) summer vs fall

(e) summer vs winter (f) fall vs swinter

Figure 5-14: Place recognition across seasons on the Nordland dataset. It can be seen that conv4 and conv5 perform better than the others, while fc6 and fc7 are the worst ($d_s = 6$).

be seen that in summer vs fall case, the performances obtained from the six layers (conv4, conv5, relu5, pool5, fc6 and fc7) are excellent (around 80% recall at 100% precision level).

For the other cases, conv4, conv5, relu5 and pool5 are more robust considering appearance changes than the higher layers fc6 and fc7.

UTBM-2 dataset: morning vs afternoon

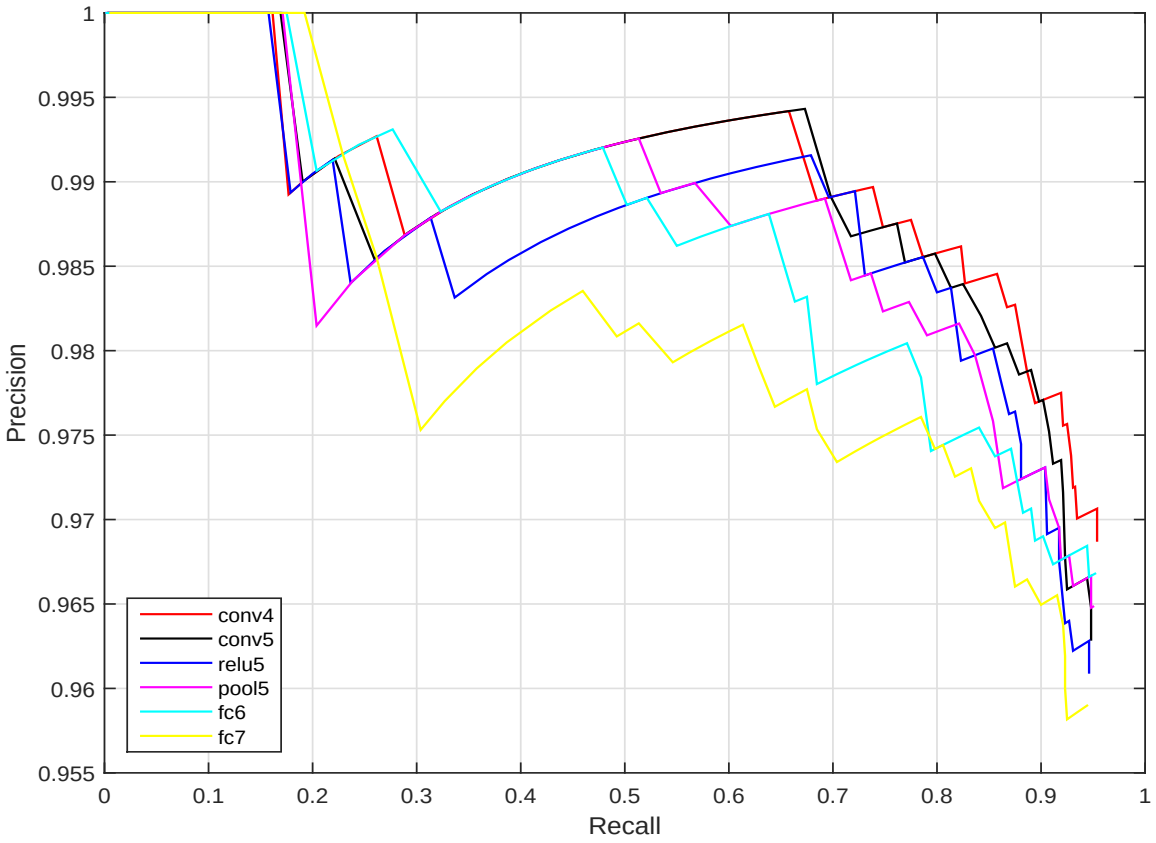


Figure 5-15: Precision-recall curves for UTBM-2 dataset considering different layers ($d_s = 8$).

Illumination change robustness

Illumination is another important factor for visual recognition. We investigate the ConvNet features performance on UTBM-2 dataset that considers morning versus afternoon situa-

tions. The precision-recall curves are presented in Fig.5-15. According to this figure, the features from mild-layers (conv4, conv5, relu5, pool5) can slightly deal with the illumination changing problem, while the other layer features (fc6 and fc7) can not deal with the severe illumination changes. This is maybe because the pre-trained network build the model under good quality images (normal illumination situation), and then it does not show strong robust ability in illumination variance situation.

According to the precision-recall curves for appearance and illumination changing situation on these four different tested datasets. It can be found that, the mid-level features from layers conv4 and relu5 are more robust against appearance and illumination changes than the other layer features. While higher layers (fc6 and fc7) in the feature hierarchy lack robustness and exhibit inferior place recognition performance.

Table 5.3 shows F_1 scores obtained for the proposed method using different layers, the previous proposed methods (D-CSLBP++HOG and CSLBP++GIST) and state-of-the-art methods like FAB-MAP and SeqSALM. For SeqSLAM comparison, the OpenSeqSLAM code [72] was used and the same sequence lengths were taken as settled above. While the other parameters are set to default values as reported in [72]. For FAB-MAP comparison, the OpenFABMAP code [38] was used. It can be found that localized sequence matching using features extracted by layer conv4 matches or exceeds the performance of the other methods.

Table 5.3: F_1 scores considering different AlexNet layers,the previous proposed methods (D-CSLBP++HOG (approach in Chapter 3) and CSLBP++GIST (approach in Chapter 4)) and other state-of-the-art methods (SeqSLAM, FAB-MAP). The † means the F_1 score is smaller than 0.1.

Dateset		Alex Layers						SeqSLAM	FAB-MAP	D-CSLBP++HOG	CSLBP++GIST
		conv4	conv5	relu5	pool5	fc6	fc7				
Nordland	spring vs summer	**0.8967**	0.8427	0.8734	0.8354	0.6722	0.5455	0.8010	†	—	0.8757
	spring vs fall	**0.8984**	0.8572	0.8821	0.8579	0.7098	0.5859	0.8569	†	—	0.8910
	spring vs winter	**0.9255**	0.8987	0.8983	0.8750	0.4795	0.2387	0.8362	†	—	0.8683
	summer vs fall	**0.9396**	0.9381	0.9388	0.9375	0.9286	0.9047	0.8435	†	—	0.9046
	summer vs winter	**0.9245**	0.8935	0.8581	0.8497	0.4142	0.1817	0.8263	†	—	0.8571
	fall vs winter	**0.9288**	0.8922	0.8598	0.8599	0.5119	0.2337	0.7360	†	—	0.8402
UTBM-1		**0.9607**	0.9576	0.9576	0.9583	0.9607	0.7762	0.8693	0.2356	0.8869	—
UTBM-2		**0.9622**	0.9564	0.9544	0.9574	0.9593	0.9516	0.8769	0.4813	0.9532	—
City Center		0.9288	0.9246	0.9264	**0.9317**	0.9299	0.9166	†	0.5326	—	—

5.4.3 LSH based visual recognition

In contrast to typical computer vision benchmarks where the recognition accuracy is the most important performance metric [69], visual localization for vehicles or robots always needs agile algorithms for real-time application. As introduced in Chapter 3, Locality Sensitive Hashing (LSH) is arguably the most popular feature compression method and has been applied to many problems, including information retrieval and computer vision [111]. Therefore, in order to speed up image matching significantly, LSH method that preserves the euclidean similarity [18] is considered again for faster visual recognition.

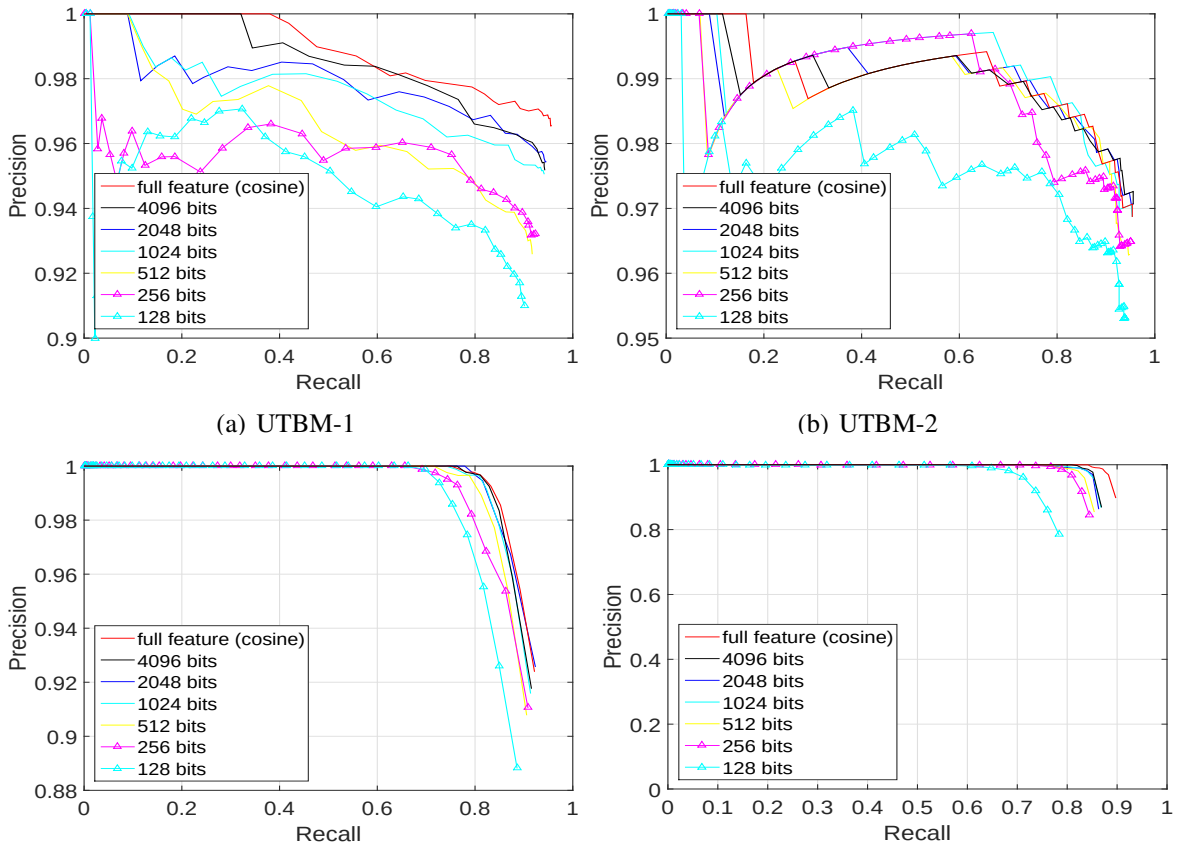
According to the study results of Section 5.4.2, conv4 has shown its strong ability in place recognition. However, computing the cosine distance between many 64,896 dimensional conv4 feature is an expensive operation. For real-time place recognition, Locality-Sensitive Hashing (LSH) method is used, mapping the conv4 feature f_{conv4} to a low-dimensional binary vector:

$$H(K) = sign(w^{\top} f_{conv4} + b) \tag{5.10}$$

where w is a K dimension data-independent random matrix, which is satisfy a standard Gaussian distribution [26]. And b is a random intercept. In our experiment, conv4 feature f_{conv4} is normalized with zero mean, then approximately balanced partition is obtained with $b = 0$. Thus, high dimensional feature is converted into a low K dimension binary bits. The binary bit vectors can then be compared using hamming distance more efficiently.

We implement this method and compare the place recognition performance achieved with the hashed conv4 feature vectors of different lengths ($2^7 \ldots 2^{12}$ bits) on the four datasets (Shown in Fig.5-16). Hashing the original 64,896 dimensional vectors into 4096 bits corresponds to a data compression of 63.1%. In addition, the 4096 hash bits representation retains approximately 95% of the original place recognition performance. It can be seen from Fig.5-16 that, when the length of hash bits is decreasing, the place recognition performance is also descending.

Table 5.4 shows the F_1 scores of different hash bit lengths achieved in four datasets. The average times per matching are also presented. The experiments are conducted on a



(a) UTBM-1 (b) UTBM-2

(c) City Center (d) Nordland (spring vs winter)

Figure 5-16: Precision-recall curves of different hash bit length. The cosine distance over the full feature vector of 64896 dimensions (red) can be closely approximated by the hamming distance over bit vectors of length 4096 (dark) without losing much performance. This corresponds to a compression of 63.1%.

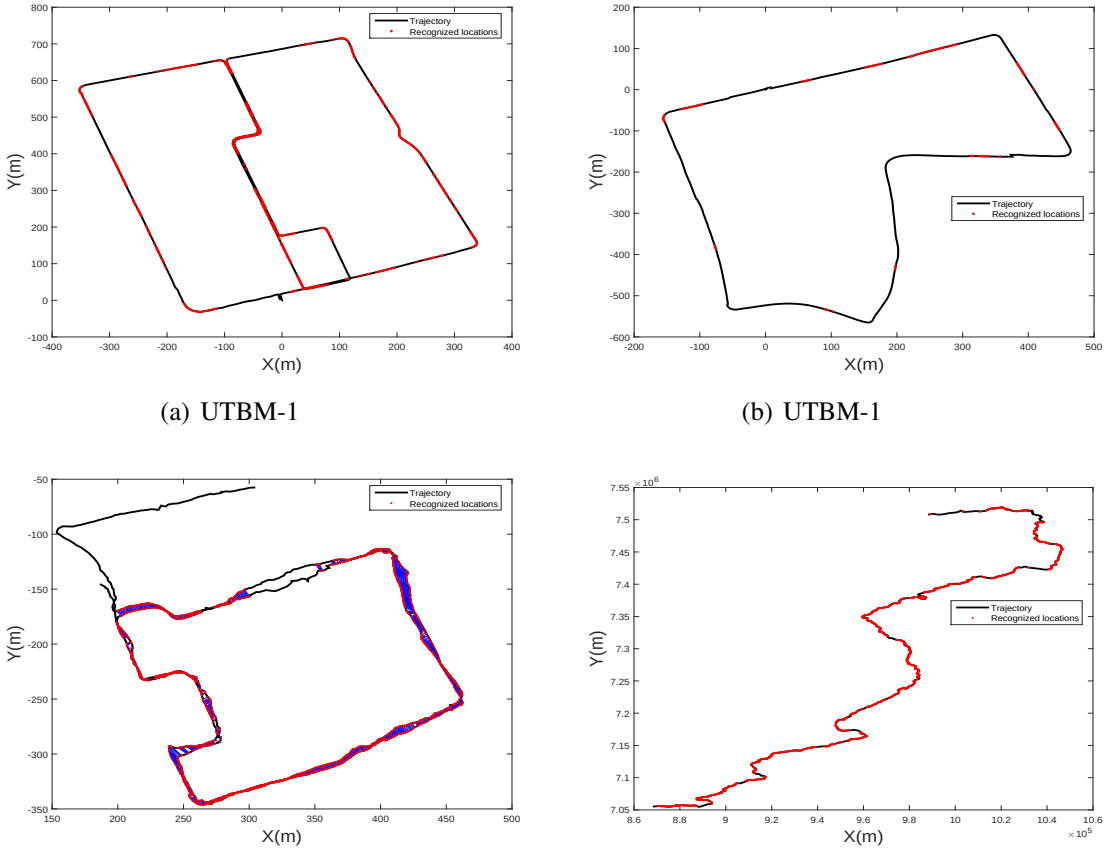
Table 5.4: F_1 scores and matching time comparison of different lengths of hash bits.

Method	F_1 score				Average time per matching(All datasets)
	UTBM-1	UTBM-2	City Center	Nordland (spring vs Winter)	
256 hash bits	0.9276	0.9574	0.9094	0.8817	0.0135 s
512 hash bits	0.9219	0.9554	0.9084	0.8944	0.0147s
1024 hash bits	0.9460	0.9612	0.9162	0.9046	0.0170s
2048 hash bits	0.9497	0.9632	0.9246	0.9064	0.0209s
4096 hash bits	0.9478	0.9641	0.9166	0.9099	0.0291s
Full feature	0.9607	0.9622	0.9228	0.9255	0.3259 s

laptop machine with intel i7-4700MQ CPU and 32G RAM. Compared with the full feature matching, hashing the original full feature into 4096 bits makes the matching easier and faster.

As shown in Table 5.4, the average time per matching using 4096 hash bits is 0.0291s which corresponds almost to a speed-up factor of 12 compared to using the cosine distance over the original conv4 feature requiring 0.3259s per matching. There is no doubt that for larger scale datasets, the speed up advantages can be more significant.

5.4.4 Visual localization results



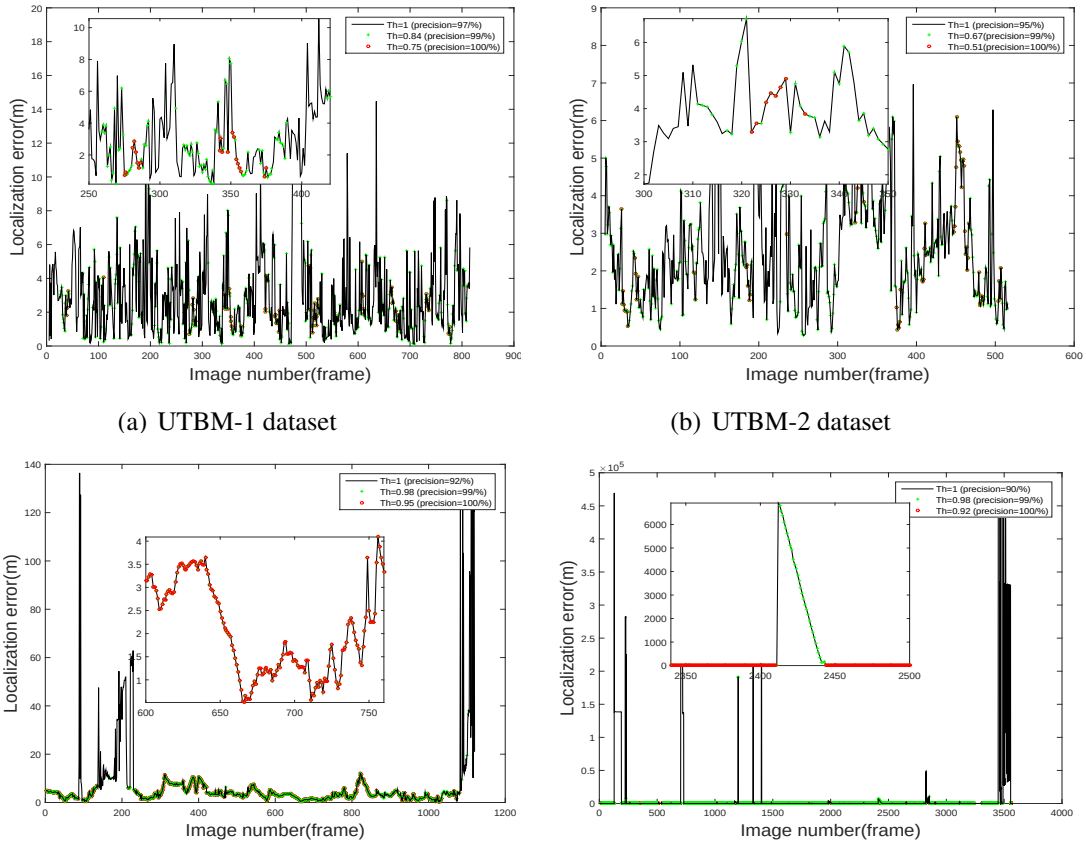
(a) UTBM-1 (b) UTBM-1

(c) City Center (d) Nordland (spring vs winter)

Figure 5-17: Visual localization results obtained by our system in the four datasets. The feature used here is 4096 hash bits of conv4 layer. Two images coming from the same location (on the basis of appearance alone) are marked with red point and joined with a blue line.

For the visual localization based on place recognition, the recognition rate at high pre-

cision level is a key indicator to reflect whether the system is enough robust to determine position under changing environment. A correct place recognition means a successful visual localization while an incorrect place recognition could cause huge localization error. Therefore, the higher the recognition rate at 100% precision is, the more robust visual localization system is. Fig.5-17 shows the final place recognition based visual localization results for the different datasets at a precision level of 100%. Regardless of the appearance and illumination changes, the proposed method can still localize the vehicle in most places. The visual localization errors at different precision levels are illustrated in Fig.5-18.



(a) UTBM-1 dataset

(b) UTBM-2 dataset

(c) City Center

(d) Nordland (spring vs winter)

Figure 5-18: Localization error at different recognition precision levels. Dark line is the localization error for all recognition results (including the true positives and false positives). Red "o" is localization error for the 100% precision level. Green "+" is the localization error for 99% precision level.

Through changing the threshold value (Th), different recognition precision levels can be recorded. At 100% precision level, all the recognized places are true positives, the localiza-

tion error is small. For achieving 100% recognition precision level, the threshold value is set to 0.75 and 0.51 for UTBM-1 and UTBM-2 datasets respectively. While for City Center and Nordland (spring vs winter) datasets, the threshold is set to 0.95 and 0.92 respectively. By increasing the threshold value, the precision is decreasing. When the threshold is set to 1, the precision level is lowest and there are many false matching for place recognition. Some false matching will lead to huge localization error, because the places can be wrongly matched to any place in the whole trajectory. In general, smaller the threshold is, fewer false recognition cases occur.

In Table 5.5, the average localization errors and recall ratio at different precision levels are given. Using 4096 hash bits, at 100% precision, the proposed approach achieves above 75% recall on the City Center dataset, and above 72.88% on the more challenging Nordland dataset (Spring vs winter). While on the UTBM-1 and UTBM-2 datasets, the recall are 32.88% and 11.54% respectively. For all these datasets, the average visual localization errors at 100% recognition precision are below than 4m. It also can be noted that, the average visual localization error is increasing with the recognition precision decreasing.

Table 5.5: Recall results and average localization error at two precision levels (4096 hash bits).

Dateset	100% precision		99% precision	
	Recall (%)	Error (/m)	Recall (%)	Error (/m)
UTBM-1	32.88	2.32	89.62	2.67
UTBM-2	11.54	2.39	51.89	2.62
City Center	76.29	3.36	82.89	3.88
Nordland (spring vs winter)	72.88	2.56	82.95	3.18

5.5 Conclusion and Future Works

Along this chapter a visual vehicle localization approach based on ConvNet features and localized sequence matching is presented. The approach takes the strengths of deep conventional network and localized sequence matching, which make place recognition fast and

accurate. The proposed visual localization approach allows the vehicle to localize itself in the most places in changing environments.

We conduct experiments on four typical datasets that consider big challenges in visual place recognition: appearance, viewpoint and illumination changes. The experimental results show that the use of ConvNet conv4 feature can obtain F_1 score above 0.85 in these four datasets, which outperforms the methods of D-CSLBP++HOG (approach in Chapter 3) and CSLBP++GIST (approach in Chapter 4), FAB-MAP and SeqSLAM. In addition, for satisfying real-time constraints, speed-up approach based on LSH method was used to compress the high dimension ConvNet feature. By using the 4096 hashing bits representation instead of the original conv4 feature, the average time per matching is almost 12 times faster.

Although ConvNet features can improve visual localization, using pre-trained network still can be improved for visual localization because the original network is trained for object classification rather than place recognition. In future work, how to train visual localization based networks and features optimizing method specifically for life-long visual localization under changing conditions will be investigated.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

This thesis focuses on improving place recognition for visual localization in changing outdoor environment. For many applications, it is crucial that a robot localizes itself within the world for autonomous navigation and driving. After a review of some state-of-art place recognition and visual localization technologies in Chapter 2, new methods to perform visual localization in changing environments were proposed in this thesis.

Firstly, multi-feature combination for vehicle localization is explored in Chapter 3. Since different types of features have their own advantages, combining some powerful features will be helpful in place recognition. We firstly integrate disparity information into complete center-symmetric local binary patterns (CSLBP) to obtain a robust global image description (D-CSLBP). Furthermore, D-CSLBP and HOG features are combined together to strengthen the place describing ability by taking the advantage of depth, texture and shape information.

The multi-feature (D-CSLBP++HOG) improves visual recognition performance, it thus allows decreasing the effect of some typical problems in place recognition such as perceptual aliasing. In addition, for real-time visual localization, local sensitive hashing method (LSH) is used to speed up the process of image matching. Our approach allows more effective visual localization compared with the state-of-the-art FAB-MAP (Fast Appearance Based Mapping) method.

Secondly, visual localization across seasons based on sequence matching and feature combination of GIST and CSLBP is developed in Chapter 4. Matching places by considering sequences instead of single images denotes high robustness to extreme perceptual changes. The proposed method is tested and evaluated in four seasons outdoor environments. Studies of the relationship between image sequence length and sequences matching based place recognition performance is conducted. The achieved results have shown improved precision-recall performance and the proposed approach outperformed the state-of-the-art SeqSLAM (Sequence Simultaneous Localization and Mapping) algorithm.

Thirdly, all-environment visual localization system based on ConvNet features and localized sequence matching is proposed in Chapter 5. We use the automatic learned Convolutional Network (ConvNet) features and localized matching technique to accomplish all-environment visual localization task under appearance and illumination changing situations. The pre-trained networks provided by MatConvNet are used to extract ConvNet features and then a localized sequence search technique is applied for visual localization. Furthermore, a comprehensive performance comparison of different ConvNet layers (each defining a level of features) is conducted on four real world datasets. The F_1 scores attained with the conv4 of ConvNet feature on the different datasets are higher than 0.85, which are significant better than those of FAB-MAP and SeqSLAM in presence of appearance and illumination changes. To speed up the computational efficiency, locality sensitive hashing method is applied to achieve real-time visual localization with minimal accuracy degradation.

6.2 Future Works

In the author's point of view, there are many ways in which the research presented in this thesis could be developed in the future.

- More metric information can be used for visual localization in further research. In this thesis, metric information was not used just because we concentrate on feature comparison and image matching under changing environment. In fact, visual localization performance could be improved considerably by integrating real vehicle

motion model. Operating on relative geometric relations between poses along the trajectory during data acquisition, metric map representations (feature maps or occupancy grid maps) of the environment can be geometric and clearly link with the real world. The localization can be done continuously and highly accurate locally.

- In terms of visual place recognition, how to describe a place appropriately is the key point. Therefore, some powerful and robust image descriptors can be developed for place recognition. Most of visual place description techniques can be classified into two broad categories: local feature descriptors and global or whole-image descriptors. Global descriptors are normally pose dependent and very fast to compute but less robustness to occlusion and perceptual aliasing effect. While local features hold a strong discriminative power but with high cost of computation time and complex match processing. How to describe a place with “locally " and “globally" information and take the advantage of the these information can be further studied.

- Three-Dimensional (3D) Information can be considered in visual localization. Unlike the 2D geometry of interest points in an image, 3D geometry information in space is an invariant property of a location. It should be possible to integrate this relative 3D distance information to reduce false place recognition between some similar locations.

- Research in place recognition can also benefit from the ongoing research in object detection and scene classification. Based on object detection, moving objects such as pedestrians or cars can be ignored while other landmark objects such as buildings can be used for long-term place recognition. Semantic scene context can furthermore limit the search space for place recognition to ensure scalability towards long-term autonomy. Semantic context can support learning and predicting the changes in a scene and help to increase robustness against environmental condition changes. Exploiting knowledge about which objects are dynamic or static and object semantic information can increase the robustness to appearance changing. Therefore, using object detection and scene classification for the task of place recognition is a worthwhile direction for future research.

- In the last, convolutional neural networks have emerged as powerful image representations tool for various category-level recognition tasks such as scene recognition, object detection and classification. However, direct use of ConvNet representation trained for object classification as "black-box" feature extractor can not achieve significant improvements in performance on instance-level recognition tasks. Another problem is that, ConvNet parameters learning is complex and hard, which restrict the real-time place recognizing for visual localization. Therefore, some flexible and easy-training deep learning networks should be developed for place recognition based visual localization.

List of Figures

1-1 Example of participants in 2007 Grand Challenge 8

1-2 Google prototype driverless car. [3] 8

1-3 Synopsis of place recognition based visual localization system. 11

1-4 Experimental vehicle equipped with sensors (especially camera and RTK-GPS). 14

2-1 General scheme of visual place recognition system, consisting of five core components. Incoming visual data is processed by place describing module. Robot's knowledge of the world is stored in place remembering module. Place recognizing module decides whether the current visual data matches a previously stored place. Throughout the place recognition process, robot localizes itself thanks to matching with a previously-visited location. Since the previously-visited location tagged with GPS information, the vehicle or robot can achieve its localization by assimilating its position with the one of the retrieved/matched image. 18

2-2 SIFT extracts interest points in an image for description. The circles are interest points selected by SIFT within the image. The number of possible features may vary depending on the number of interest points detected in the image. 22

2-3 An example of detecting SURE features in depth images at locations with locally prominent surface curvature (from [34]). SURE feature captures local shape and colored texture at interest points. Based on SURE features, places are recognized using a bag-of-words approach. 28

2-4 Place recognition system utilizing convolutional network features as robust landmark descriptors to recognize places despite severe viewpoint and condition changes, without requiring any environment-specific training. The colored boxes in the images above show ConvNet landmarks that have been correctly matched between two significantly different viewpoints of a scene. This enabling place recognition under challenging conditions (from [102]). 30

2-5 Example of place recognition based on single image. Each testing image has to retrieve its similar one from the training database. 32

2-6 An example of place recognition based on image sequence. Sequences A and A' are taken in a time interval of two weeks. 33

3-1 Multi-feature built from gray-scale image and disparity map. Features are firstly extracted from each image block and then concatenated together. The symbol "++" means concatenation. 39

3-2 Illustration of the basic LBP operator 41

3-3 Examples of (P,R) neighborhood used to compute LBP: (8,1), (16,2) and (8,2) 42

3-4 Example of HOG feature. 48

3-5 The process of the proposed place recognition based visual localization. . . 49

3-6 Image preprocessing using contrast-limited adaptive histogram equalization(CLAHE). The left image is processed using CLAHE and the prepossessing result is the right image. 50

3-7 An example of block based local binary descriptor extraction. Features are extracted from each image block firstly and then concatenated together. Here, image blocks are non-overlapped and do not need any image segmentation. 51

3-8 Block based feature extraction procedure (applied to each images pair of the training database for the off-line phase and to current images pair for the testing phase) 52

3-9 Vehicle paths for the UTBM-1 and UTBM-2 datasets. Source: Google Maps 58
3-10 Example of gray-scale image and its corresponding local binary images (LBP, CLBP, CSLBP, CSLDP and XCSLBP) and HOG feature. 59
3-11 Image retrieval performance (precision-recall curve) comparison considering different block based binary features on UTBM-1 and KITTI 06 datasets. Here the image block size is 32. 62
3-12 Image retrieval performance (precision-recall curve) comparison considering D-CSLBP feature extracted with different image block sizes, on four datasets. 63
3-13 Image retrieval performance (precision-recall curve) comparison of HOG, D-CSLBP, D-CSLBP++HOG based approaches, on four datasets. 64
3-14 Example of distance matrices for UTBM-1 dataset. Here, the distance matrix D is normalized into 0-1 range. The distances of same or similar images are close to 0 (red color), while for the larger distances, the corresponding color is close to yellow. In (a), two images from a same place are taken at different times (difference of two weeks). From figure (b) to figure (f), the distance matrix D is plotted. The distance matrices show that multi-feature combination (c) reduces the noise appeared around the diagonal (ground-truth line). Besides, compared with (d), after adding disparity information in (e), perceptual aliasing decreases, as confirmed by the precision-recall curves in Fig.3-13 (a). 66
3-15 Image retrieval performance (precision-recall curve) comparison of different hash bit lengths. 68
3-16 Visual localization results obtained by our system on four datasets. The trajectory of the vehicle is depicted with black lines, the loop closure zone is plotted by blue lines. Red points are correctly recognized locations at 100% precision by using our proposed approach. There are no false positives in any case. It is noted that the loop closure zone of datasets UTBM-1 and UTBM-2 is the whole trajectory, while the loop closure zone of KITTI 05 and KITTI 06 is only parts of the trajectory in blue. 70

3-17 Place recognition based localization errors at different precision levels. The dark line is the localization error for all recognition results (including the true positives and false positives). Red "o" is localization error at 100% precision level. Green "+" is localization error at 99% precision level. Blue "." is localization error at 90% precision level. 71

3-18 An example of LiDAR localization results. 72

4-1 General diagram of visual localization system using sequence matching. . . 77

4-2 Flow chart of proposed visual localization using sequence matching. 78

4-3 Example of extracted features. The first row shows the original images. The second and third row show the images of LBP and CSLBP features respectively. The fourth row gives the images of GIST features. 82

4-4 Nordland route. Source: Google map. The trajectory is recorded four times, once in every season. Video sequences are synchronized and the camera position and field of view are always the same. GPS readings are available. 86

4-5 A typical four seasons images representing the same scene in spring, summer, fall and winter. It can be seen that show huge differences appear in the images with season changing. 87

4-6 Performance of the proposed method according to different used features and comparison with SeqSLAM method (summer vs winter, sequence length L_{length}=200). 89

4-7 Performance comparison of our proposed method with different feature combination, according to image sequence length L_{length} (spring vs winter). 90

4-8 Ground-truth. Since frame from one season corresponds to the same frame in any of the other three seasons, the ground-truth is diagonal. 91

4-9 Matching results under different season couples at 100% recall situation. The expected result is along the diagonal. 91

4-10 Precision-recall curves comparing the performance of different feature combination under different season couples. 93

4-11 Corresponding frames from sequence matching between two seasons (fall and winter). The left column shows fall image frames queried from winter traversal, and the right column shows the winter image frames recalled by our approach. 94

4-12 Visual localization results under different season couples at 100% precision level. Successful matched images that come from the same location (on the basis of appearance alone) are marked in red points. 96

5-1 Schematic illustration of visual localization using ConvNet features and localized sequence matching. ConvNet features are extracted from testing images and then compared to those extracted from all images of the training database. After feature comparison, localized sequence matching is conducted to find the best image matching. 101

5-2 Detailed block diagram of the proposed visual localization method. Feature extraction uses pre-trained network, feature comparison uses cosine distance and localized sequence searching is based on potential paths. . . . 102

5-3 Architecture of the caffe-alex network. ConvNet transforms the original image, layer by layer, from the original pixel values to the final class scores. 105

5-4 An example of a scene and extracted features from different layers of the caffe-alex network. Features obtained from different ConvNet layers can serve as holistic image descriptors for place describing. 106

5-5 Procedure for ConvNet features comparison. Features are extracted from each testing image and then compared with features extracted from all training images. 108

5-6 The search algorithm finds the lowest-cost straight line within the searching matrix M^T. These lines are the set of potential paths through the matrix. The red line is the lowest-cost path which aligns the testing sequence and training sequence. Each element indicates the cosine distance between two images (test and train). 109

5-7 UTBM-1 dataset: Example of training and testing images (interval time of one week). Left column is training images and right column is testing images. 113

5-8 UTBM-2 dataset: morning vs afternoon. Left column is training images and right column is testing images. 113

5-9 City Center dataset: twice traveling. Left column is training images and right column is testing images. 114

5-10 Two examples of performance comparison of our proposal depending on the image sequence length (d_s) in the challenging UTBM-1 and Nordland (fall vs winter) datasets. The feature used here is conv4 layer. 116

5-11 Examples of frame matches from the Nordland dataset (fall vs winter). The top row shows testing frames, and the middle and third rows show the training frames recalled by $d_s = 1$ (single image) and $d_s = 6$, respectively. Visual recognition based on sequence matching achieves better performance than those obtained using single image. 117

5-12 Precision-recall curves for UTBM-1 dataset (the trajectory acrosses forest, city and parking areas)($d_s = 8$). 118

5-13 Precision-recall curves for City Center dataset (the trajectory acrosses city and parking areas)($d_s = 3$). 119

5-14 Place recognition across seasons on the Nordland dataset. It can be seen that conv4 and conv5 perform better than the others, while fc6 and fc7 are the worst ($d_s = 6$). 120

5-15 Precision-recall curves for UTBM-2 dataset considering different layers ($d_s = 8$). 121

5-16 Precision-recall curves of different hash bit length. The cosine distance over the full feature vector of 64896 dimensions (red) can be closely approximated by the hamming distance over bit vectors of length 4096 (dark) without losing much performance. This corresponds to a compression of 63.1%. 124
5-17 Visual localization results obtained by our system in the four datasets. The feature used here is 4096 hash bits of conv4 layer. Two images coming from the same location (on the basis of appearance alone) are marked with red point and joined with a blue line. 125
5-18 Localization error at different recognition precision levels. Dark line is the localization error for all recognition results (including the true positives and false positives). Red "o" is localization error for the 100% precision level. Green "+" is the localization error for 99% precision level. 126

List of Tables

2.1 Summary of main local features 21

2.2 Summary of main global features 24

3.1 F_1 score comparison for different tested binary features on four datasets. Here the block size is set to 32×32. 62

3.2 Comparison of F_1 scores for different features and the state-of-the-art FAB-MAP method, on four datasets. 65

3.3 F_1 score and matching times comparison of different hash bit lengths for our approach and the state-of-the-art FAB-MAP method. 69

3.4 Recall results and average localization error at three precision levels (4096 hash bits). 72

4.1 Recall scores at selected high precision levels (100%, 99%, 90%) 95

4.2 Comparison of average processing times for image matching(/s) 97

5.1 The layers from the caffe-alex ConvNet model used in our evaluation and their output dimensionality (height×width×feature map number). 107

5.2 Description of the main characteristics of the datasets employed in the experiments. 112

5.3 F_1 scores considering different AlexNet layers,the previous proposed methods (D-CSLBP++HOG (approach in Chapter 3) and CSLBP++GIST (approach in Chapter 4)) and other state-of-the-art methods (SeqSLAM, FAB-MAP). The † means the F_1 score is smaller than 0.1. 122

5.4 F_1 scores and matching time comparison of different lengths of hash bits. . 124

5.5 Recall results and average localization error at two precision levels (4096 hash bits). 127

Bibliography

[1] A. Alahi, R. Ortiz, and P. Vandergheynst. FREAK: Fast retina keypoint. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 510–517, June 2012.

[2] Pablo F Alcantarilla and TrueVision Solutions. Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans. Patt. Anal. Mach. Intell*, 34(7):1281–1298, 2011.

[3] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison. KAZE features. In *European Conference on Computer Vision*, pages 214–227. Springer, 2012.

[4] H. Andreasson and T. Duckett. Topological localization for mobile robots using omni-directional vision and local features. In *5th IFAC Symposium on Intelligent Autonomous Vehicles*, Lisbon, Portugal, 2004.

[5] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, June 2016.

[6] R. Arroyo, P.F. Alcantarilla, L.M. Bergasa, J.J. Yebes, and S. Bronte. Fast and effective visual place recognition using binary codes and disparity information. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3089–3094, Sept 2014.

[7] R. Arroyo, P.F. Alcantarilla, L.M. Bergasa, J.J. Yebes, and S. Gamez. Bidirectional loop closure detection on panoramas for visual navigation. In *IEEE Intelligent Vehicles Symposium Proceedings*, pages 1378–1383, June 2014.

[8] H. Badino, D. Huber, and T. Kanade. Real-time topometric localization. In *IEEE International Conference on Robotics and Automation*, pages 1635–1642, May 2012.

[9] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. *SURF: Speeded Up Robust Features*, pages 404–417. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[10] D. Bekele, M. Teutsch, and T. Schuchert. Evaluation of binary keypoint descriptors. In *IEEE International Conference on Image Processing*, pages 3652–3656, Sept 2013.

[11] Yoshua Bengio et al. Learning deep architectures for AI. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[12] R. Bishop. *Intelligent Vehicle Technology and Trends*. Artech House ITS library. Artech House, 2005.

[13] S. Brahnam, L. Jain, A. Lumini, and L. Nanni. Introduction to local binary patterns: New variants and applications. In *Local Binary Patterns: New Variants and Applications*, pages 1–13. Springer, 2014.

[14] C. Cadena, D. Galvez-López, F. Ramos, J. D. Tardos, and J. Neira. Robust place recognition with stereo cameras. In *IEEE International Conference on Intelligent Robots and Systems*, pages 5182–5189, Oct 2010.

[15] C. Cadena, D. Galvez-López, J. D. Tardos, and J. Neira. Robust place recognition with stereo sequences. *IEEE Transactions on Robotics*, 28(4):871–885, Aug 2012.

[16] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. BRIEF: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792. Springer, 2010.

[17] D. Caruso, J. Engel, and D. Cremers. Large-scale direct slam for omnidirectional cameras. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 141–148, Sept 2015.

[18] Moses S Charikar. Similarity estimation techniques from rounding algorithms. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, pages 380–388. ACM, 2002.

[19] Siyu Chen, Libo Huang, Jie Bai, Haitao Jiang, and Liang Chang. Multi-sensor information fusion algorithm with central level architecture for intelligent vehicle environmental perception system. Technical report, SAE Technical Paper, 2016.

[20] Yan Chen, Yingju Shen, Xin Liu, and Bineng Zhong. 3D object tracking via image sets and depth-based occlusion detection. *Signal Processing*, 112:146–153, 2015.

[21] Zetao Chen, A. Jacobson, U.M. Erdem, M.E. Hasselmo, and M. Milford. Multi-scale bio-inspired place recognition. In *IEEE International Conference on Robotics and Automation*, pages 1895–1901, May 2014.

[22] Zetao Chen, Obadiah Lam, Adam Jacobson, and Michael Milford. Convolutional Neural Network-based place recognition. In *Australasian Conference on Robotics and Automation 2014*, The University of Melbourne, Victoria, Australia, December 2014.

[23] J. Collier, S. Se, and V. Kotamraju. Multi-sensor appearance-based place recognition. In *International Conference on Computer and Robot Vision*, pages 128–135, 2013.

[24] Mark Cummins and Paul Newman. FAB-MAP: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

[25] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893, June 2005.

[26] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*, pages 253–262. ACM, 2004.

[27] A. J. Davison and D. W. Murray. Simultaneous localization and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):865–880, Jul 2002.

[28] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, June 2007.

[29] F. Dos Santos, P. Costa, and A.P. Moreira. A visual place recognition procedure with a markov chain based filter. In *IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, pages 333–338, 2014.

[30] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '09, pages 19:1–19:8, 2009.

[31] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard. An evaluation of the RGB-D SLAM system. In *IEEE International Conference on Robotics and Automation*, pages 1691–1696, May 2012.

[32] Ryan Eustice, Hanumant Singh, John Leonard, Matthew Walter, and Robert Ballard. Visually navigating the rms titanic with slam information filters. In *Proceedings of Robotics: Science and Systems (RSS)*, pages 57–64, Cambridge, MA, June 2005.

[33] R Finman, L Paull, and J J Leonard. Toward object-based place recognition in dense RGB-D maps. *ICRA workshop on visual place recognition in changing environments*, Oct 2015.

[34] T. Fiolka, J. Stĺźckler, D. A. Klein, D. Schulz, and S. Behnke. Distinctive 3D surface entropy features for place recognition. In *European Conference on Mobile Robots*, pages 204–209, Sept 2013.

[35] Emilio Garcia-Fidalgo and Alberto Ortiz. Vision-based topological mapping and localization methods: A survey. *Robotics and Autonomous Systems*, 64:1–20, 2015.

[36] J. Gaspar, N. Winters, and J. Santos-Victor. Vision-based navigation and environmental representations with an omnidirectional camera. *IEEE Transactions on Robotics and Automation*, 16(6):890–898, Dec 2000.

[37] A. Gil, O. Reinoso, O. M. Mozos, C. Stachniss, and W. Burgard. Improving data association in vision-based SLAM. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2076–2081, Oct 2006.

[38] A. Glover, W. Maddern, M. Warren, S. Reid, M. Milford, and G. Wyeth. OpenFABMAP: An open source toolbox for appearance-based loop closure detection. In *IEEE International Conference on Robotics and Automation*, pages 4730–4735, May 2012.

[39] Arren Glover, Edward Pepperell, Gordon Wyeth, Ben Upcroft, and Michael Milford. Repeatable condition-invariant visual odometry for sequence-based place recognition. In *Proceedings of the Australasian Conference on Robotics and Automation (ACRA)*, 2015.

[40] Toon Goedemé, Marnix Nuttin, Tinne Tuytelaars, and Luc Van Gool. Markerless computer vision based localization using automatically generated topological maps. In *European Navigation Conference*, 2004.

[41] Ruben Gomez-Ojeda, Manuel Lopez-Antequera, Nicolai Petkov, and Javier González Jiménez. Training a convolutional neural network for appearance-invariant place recognition. *arXiv preprint arXiv:1505.07428*, abs/1505.07428, 2015.

[42] Z. Guo, L. Zhang, and D. Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE Transactions on Image Processing*, 19(6):1657–1663, June 2010.

[43] Marko Heikkilä, Matti Pietikäinen, and Cordelia Schmid. Description of interest regions with center-symmetric local binary patterns. In *Computer Vision, Graphics and Image Processing*, pages 58–69. Springer, 2006.

[44] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[45] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM International Conference on Multimedia*, pages 675–678, New York, NY, USA, 2014.

[46] E. Johns and G.Z. Yang. Dynamic scene models for incremental, long-term, appearance-based localisation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2731–2736, 2013.

[47] E. Johns and G.Z. Yang. Feature co-occurrence maps: Appearance-based localisation throughout the day. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3212–3218, 2013.

[48] Aram Kawewong, Sirinart Tangruamsub, and Osamu Hasegawa. Position-invariant robust features for long-term recognition of dynamic outdoor scenes. *IEICE Transactions on Information and Systems*, 93(9):2587–2601, 2010.

[49] Yan Ke and R. Sukthankar. PCA-SIFT: a more distinctive representation for local image descriptors. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–506–II–513, June 2004.

[50] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 225–234, Nov 2007.

[51] Jan Knopp, Josef Sivic, and Tomas Pajdla. *Avoiding Confusing Features in Place Recognition*, pages 748–761. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[52] K. Konolige and M. Agrawal. Frameslam: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, 24(5):1066–1077, Oct 2008.

[53] J. Kosecka, Liang Zhou, P. Barber, and Z. Duric. Qualitative image based localization in indoors environments. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages II–3–II–8 vol.2, June 2003.

[54] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. In F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, pages 1097–1105. 2012.

[55] Benjamin Kuipers. The spatial semantic hierarchy. *Artificial Intelligence*, 119(1):191 – 233, 2000.

[56] Gustaf Kylberg and Ida-Maria Sintorn. Evaluation of noise robustness for local binary pattern descriptors in texture classification. *EURASIP Journal on Image and Video Processing*, 2013(1):1–20, 2013.

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

[58] Endre László, Péter Szolgay, and Zoltán Nagy. Analysis of a GPU based CNN implementation. In *13th International Workshop on Cellular Nanoscale Networks and their Applications*, pages 1–5. IEEE, 2012.

[59] H. Lategahn, J. Beck, B. Kitt, and C. Stiller. How to learn an illumination robust image feature for place recognition. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 285–291, June 2013.

[60] Y LeCun, B Boser, J Denker, D Henderson, R Howard, W Hubbard, and L Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, Dec 1989.

[61] S. Leutenegger, M. Chli, and R. Y. Siegwart. BRISK: Binary robust invariant scalable keypoints. In *2011 International Conference on Computer Vision*, pages 2548–2555, Nov 2011.

[62] Mathias Lidberg and TJ Gordon. Automated driving and autonomous functions on road vehicles. *Vehicle System Dynamics*, 53(7), 2015.

[63] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[64] S. Lowry, N. SÍźnderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, Feb 2016.

[65] Huimin Lu, Xun Li, Hui Zhang, and Zhiqiang Zheng. Robust place recognition based on omnidirectional vision and real-time local visual features for mobile robots. *Advanced Robotics*, 27(18):1439–1453, 2013.

[66] Yang Lu, Song-Chun Zhu, and Ying Nian Wu. Learning FRAME models using CNN filters for knowledge visualization. *arXiv preprint arXiv:1509.08379*, 2015.

[67] W. Maddern, M. Milford, and G. Wyeth. Towards persistent indoor appearance-based localization, mapping and navigation using CAT-Graph. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4224–4230, Oct 2012.

[68] Will Maddern, Michael Milford, and Gordon Wyeth. CAT-SLAM: Probabilistic localisation and mapping using a continuous appearance-based trajectory. *The International Journal of Robotics Research*, 31(4):429–451, April 2012.

[69] M. Milford, S. Lowry, N. Sunderhauf, S. Shirazi, E. Pepperell, B. Upcroft, C. Shen, G. Lin, F. Liu, C. Cadena, and I. Reid. Sequence searching with deep-learnt depth for condition- and viewpoint-invariant route-based place recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 18–25, June 2015.

[70] M. J. Milford, I. Turner, and P. Corke. Long exposure localization in darkness using consumer cameras. In *IEEE International Conference on Robotics and Automation*, pages 3755–3761, May 2013.

[71] Michael Milford and Gordon Wyeth. Persistent navigation and mapping using a biologically inspired slam system. *The International Journal of Robotics Research*, 29(9):1131–1153, 2010.

[72] M.J. Milford and G.F. Wyeth. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1643–1649, May 2012.

[73] H. Morioka, S. Yi, and O. Hasegawa. Vision-based mobile robot's slam and navigation in crowded environments. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3998–4005, Sept 2011.

[74] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardĺős. ORB-SLAM: A versatile and accurate monocular slam system. *IEEE Transactions on Robotics*, 31(5):1147–1163, Oct 2015.

[75] A.C. Murillo, C. Sagüés, J.J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool. From omnidirectional images to hierarchical localization. *Robotics and Autonomous Systems*, 55(5):372 – 382, 2007.

[76] J. Neira, A. J. Davison, and J. J. Leonard. Guest editorial special issue on visual SLAM. *IEEE Transactions on Robotics*, 24(5):929–931, Oct 2008.

[77] P. Neubert, N. Sunderhauf, and P. Protzel. Appearance change prediction for long-term navigation across seasons. In *European Conference on Mobile Robots (ECMR)*, pages 198–203, Sept 2013.

[78] P. Newman, D. Cole, and K. Ho. Outdoor slam using visual appearance and laser ranging. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1180–1187, 2006.

[79] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.

[80] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, Jul 2002.

[81] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001.

[82] Edward Pepperell, Peter I Corke, and Michael J Milford. All-environment visual place recognition with SMART. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1612–1618, 2014.

[83] Etta D Pisano, Shuquan Zong, Bradley M Hemminger, Marla DeLuca, R Eugene Johnston, Keith Muller, M Patricia Braeuning, and Stephen M Pizer. Contrast limited adaptive histogram equalization image processing to improve the detection of simulated spiculations in dense mammograms. *Journal of Digital imaging*, 11(4):193–200, 1998.

[84] Yongliang Qiao, Cindy Cappelle, and Yassine Ruichek. Place recognition based visual localization using LBP feature and SVM. In *14th Mexican International Conference on Artificial Intelligence (MICAI'2015), Lecture Notes in Artificial Intelligence (LNAI), Springer, vol. 9414*, Cuernavaca, Mexico, Oct 2015.

[85] A. Ranganathan, S. Matsumoto, and D. Ilstrup. Towards illumination invariance for visual localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3791–3798, May 2013.

[86] Deepak Ravichandran, Patrick Pantel, and Eduard Hovy. Randomized algorithms and NLP: Using locality sensitive hash function for high speed noun clustering. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 622–629, Stroudsburg, PA, USA, 2005.

[87] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, CVPRW '14, pages 512–519, Washington, DC, USA, 2014. IEEE Computer Society.

[88] H. Ren and Z. N. Li. Object detection using edge histogram of oriented gradient. In *IEEE International Conference on Image Processing (ICIP)*, pages 4057–4061, Oct 2014.

[89] Nicholas Roy, Paul Newman, and Siddhartha Srinivasa. *Visual Route Recognition with a Handful of Bits*, page 504. MIT Press, 2013.

[90] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, Nov 2011.

[91] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[92] G. Schindler, M. Brown, and R. Szeliski. City-scale location recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7, June 2007.

[93] Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, and Yann LeCun. OverFeat: Integrated recognition, localization and detection using convolutional networks. In *International Conference on Learning Representations (ICLR 2014)*. CBLS, April 2014.

[94] Ivan Sikirić, Karla Brkić, and Siniša Šegvić. Classifying traffic scenes using the GIST image descriptor. *arXiv preprint arXiv:1310.0316*, 2013.

[95] Caroline Silva, Thierry Bouwmans, and Carl Frélicot. An extended center-symmetric local binary pattern for background modeling and subtraction in videos. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, VISAPP*, 2015.

[96] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings 9th IEEE International Conference on Computer Vision*, pages 1470–1477, Oct 2003.

[97] Dongjin Song and Dacheng Tao. Biologically inspired feature manifold for scene classification. *IEEE Transactions on Image Processing*, 19(1):174–184, 2010.

[98] Z. Sun, X. Wang, and J. Liu. Application of image retrieval based on the improved local binary pattern. In *Proceedings of the 4th International Conference on Computer Engineering and Networks*, pages 531–538, 2015.

[99] N. Sünderhauf and P. Protzel. BRIEF-Gist-Closing the loop by simple means. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1234–1241, Sept 2011.

[100] Niko Sünderhauf, Peer Neubert, and Peter Protzel. Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons. In *Proc. of Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[101] Niko Sünderhauf, Sareh Shirazi, Feras Dayoub, Ben Upcroft, and Michael Milford. On the performance of ConvNet features for place recognition. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4297–4304, 2015.

[102] Niko Sunderhauf, Sareh Shirazi, Adam Jacobson, Feras Dayoub, Edward Pepperell, Ben Upcroft, and Michael Milford. Place recognition with ConvNet landmarks: Viewpoint-robust, condition-robust, training-free. *Proceedings of Robotics: Science and Systems XII*, 2015.

[103] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1023–1029, 2000.

[104] B. Upcroft, C. McManus, W. Churchill, W. Maddern, and P. Newman. Lighting invariant urban street classification. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1712–1718, May 2014.

[105] Andrea Vedaldi and Karel Lenc. Matconvnet: Convolutional neural networks for MATLAB. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692, 2015.

[106] J. Wang and Y. Yagi. Robust location recognition based on efficient feature integration. In *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 97–101, Dec 2012.

[107] Min-Liang Wang and Huei-Yung Lin. A hull census transform for scene change detection and recognition towards topological map building. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 548–553, Oct 2010.

[108] Xiaoyu Wang, T.X. Han, and Shuicheng Yan. An HOG-LBP human detector with partial occlusion handling. In *IEEE 12th International Conference on Computer Vision*, pages 32–39, Sept 2009.

[109] Lijun Wei, Cindy Cappelle, and Yassine Ruichek. Camera/laser/gps fusion method for vehicle positioning under extended nis-based sensor validation. *IEEE Transactions on Instrumentation and Measurement*, 62(11):3110–3122, 2013.

[110] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor. Omni-directional vision for robot navigation. In *Proceedings IEEE Workshop on Omnidirectional Vision*, pages 21–28, 2000.

[111] Yan Xia, Kaiming He, Pushmeet Kohli, and Jian Sun. Sparse projections for high-dimensional binary codes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3332–3339, 2015.

[112] Gengjian Xue, Li Song, Jun Sun, and Meng Wu. Hybrid center-symmetric local pattern for dynamic background subtraction. In *IEEE International Conference on Multimedia and Expo*, pages 1–6, July 2011.

[113] Y. Yamauchi, C. Matsushima, T. Yamashita, and H. Fujiyoshi. Relational HOG feature with wild-card for object detection. In *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1785–1792, Nov 2011.

[114] X. Yang and K. T. T. Cheng. Local difference binary for ultrafast and distinctive feature description. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(1):188–194, Jan 2014.

[115] Olga Zoidi, Nikos Nikolaidis, Anastasios Tefas, and Ioannis Pitas. Stereo object tracking with fusion of texture, color and disparity information. *Signal Processing: Image Communication*, 29(5):573–589, 2014.

Document réalisé avec LaTeX et :
le style LaTeX pour Thèse de Doctorat créé par S. Galland — http://www.multiagent.fr/ThesisStyle
la collection de paquets tex-upmethodology — http://www.arakhne.org/tex-upmethodology/

Résumé :

Dans de nombreuses applications, il est crucial qu'un robot ou un véhicule se localise, notamment pour la navigation ou la conduite autonome. Cette thèse traite de la localisation visuelle par des méthodes de reconnaissance de lieux. Le principe est le suivant : lors d'une phase hors-ligne, des images géo-référencées de l'environnement d'évolution du véhicule sont acquises, des caractéristiques en sont extraites et sauvegardées. Puis lors de la phase en ligne, il s'agit de retrouver l'image (ou la séquence d'images) de la base d'apprentissage qui correspond le mieux à l'image (ou la séquence d'images) courante. La localisation visuelle reste un challenge car l'apparence et l'illumination changent drastiquement en particulier avec le temps, les conditions météorologiques et les saisons. Dans cette thèse, on cherche alors à améliorer la reconnaissance de lieux grâce à une meilleure capacité de description et de reconnaissance de la scène. Plusieurs approches sont proposées dans cette thèse : 1) La reconnaissance visuelle de lieux est améliorée en considérant les informations de profondeur, de texture et de forme par la combinaison de plusieurs de caractéristiques visuelles, à savoir les descripteurs CSLBP (extraits sur l'image couleur et l'image de profondeur) et HOG. De plus l'algorithme LSH (Locality Sensitive Hashing) est utilisée pour améliorer le temps de calcul ; 2) Une méthode de la localisation visuelle basée sur une reconnaissance de lieux par mise en correspondance de séquence d'images (au lieu d'images considérées indépendamment) et combinaison des descripteurs GIST et CSLBP est également proposée. Cette approche est en particulier testée lorsque les bases d'apprentissage et de test sont acquises à des saisons différentes. Les résultats obtenus montrent que la méthode est robuste aux changements perceptuels importants ; 3) Enfin, la dernière approche de localisation visuelle proposée est basée sur des caractéristiques apprises automatiquement (à l'aide d'un réseau de neurones à convolution) et une mise en correspondance de séquences localisées d'images. Pour améliorer l'efficacité computationnelle, l'algorithme LSH est utilisé afin de viser une localisation temps-réell avec une dégradation de précision limitée.

Mots-clés : localisation visuelle, reconnaissance de lieux, recherche d'images par le contenu, combinaison de caractéristiques visuelles, apprentissage profond

Abstract :

In many applications, it is crucial that a robot or vehicle localizes itself within the world especially for autonomous navigation and driving. The goal of this thesis is to improve place recognition performance for visual localization in changing environment. The approach is as follows : in off-line phase, geo-referenced images of each location are acquired, features are extracted and saved. Then, in the on-line phase, the vehicle localizes itself by identifying a previously-visited location through image or sequence retrieving. However, visual localization is challenging due to drastic appearance and illumination changes caused by weather conditions or seasonal changing. This thesis addresses the challenge of improving place recognition techniques through strengthen the ability of place describing and recognizing. Several approaches are proposed in this thesis : 1) Multi-feature combination of CSLBP (extracted from gray-scale image and disparity map) and HOG features is used for visual localization. By taking the advantages of depth, texture and shape information, visual recognition performance can be improved. In addition, local sensitive hashing method (LSH) is used to speed up the process of place recognition ; 2) Visual localization across seasons is proposed based on sequence matching and feature combination of GIST and CSLBP. Matching places by considering sequences and feature combination denotes high robustness to extreme perceptual changes ; 3) All-environment visual localization is proposed based on automatic learned Convolutional Network (ConvNet) features and localized sequence matching. To speed up the computational efficiency, LSH is taken to achieve real-time visual localization with minimal accuracy degradation.

Keywords : visual localization, place recognition, image retrieval, feature combination, deep learning

■ École doctorale SPIM - Université de Technologie Belfort-Montbéliard
F - 90010 Belfort Cedex ■ tél. +33 (0)3 84 58 31 39
■ ed-spim@univ-fcomte.fr ■ www.ed-spim.univ-fcomte.fr