Visual Place Recognition in Changing Environments Utilising Sequence-Based Filtering and Extremely JPEG Compressed Images

Mihnea-Alexandru Tomiță

A thesis submitted for the degree of

Doctor of Philosophy

at the

School of Computer Science and Electronic Engineering

University of Essex

August 2023

Declaration

I hereby declare that the content of this thesis is the result of my own independent work and that I have acknowledged all sources used in the preparation of this manuscript. I certify that this thesis is submitted in fulfillment of the requirements for the Doctor of Philosophy degree at the University of Essex.

Mihnea-Alexandru Tomiță

August 2023

Acknowledgements

I would like to express my sincerest gratitude to my supervisors, Dr. Shoaib Ehsan and Prof. Klaus McDonald-Maier, who offered me the opportunity to pursue a PhD degree and be a part of their laboratory. I am extremely grateful for their continuous support, guidance, and encouragement through this beautiful journey, that ultimately allowed me to complete my most important educational accomplishment to date. I would also like to thank Prof. Michael Milford for his insightful comments on my research, that ultimately improved the quality of this work.

I am also thankful to my colleagues in the Embedded and Intelligent Systems (EIS) group for creating a supportive and stimulating research environment. I would like to express my deepest appreciation to my colleagues Mubariz Zaffar and Bruno Ferrarini for their continuous support during my PhD. Our discussions have been a source of motivation and inspiration throughout this challenging journey. I wish them the best of luck for their future.

I am thankful to the UK Engineering and Physical Sciences Research Council for their financial support, which made this research possible.

Lastly, I would like to express my love and appreciation to my family and friends. Their continuous support and encouragement have been the foundation of my success, allowing me to pursue my academic goals. I will be forever grateful for their unconditional love.

Abstract

Visual Place Recognition (VPR), part of Simultaneous Localisation and Mapping (SLAM), is an essential task for the localisation process, where each robotic platform is required to successfully navigate through its environment using visual information gathered from the on-board camera. Despite the recent efforts of the research community, VPR remains an improving process. To this end, a large number of deep-learning-based and handcrafted VPR techniques (also referred as learnt and non-learnt VPR techniques) have been proposed to overcome the challenges in this field, such as viewpoint, illumination and seasonal variations. While Convolutional Neural Network (CNN)-based VPR techniques have significant computational requirements that may restrict their applicability on resource-constrained platforms, handcrafted VPR techniques struggle with appearance changes. In this thesis, two mainly unexplored avenues of research are investigated, namely sequence-based filtering and JPEG compression.

To overcome the previously mentioned challenges, this thesis proposes a handcrafted VPR technique based on HOG descriptors, paired with an adaptive sequence-based filtering schema to perform VPR in scenarios where the appearance of the environment drastically changes upon different traversals. The technique entitled ConvSequential-SLAM is capable of achieving comparable place matching performance with state-of-the-art VPR techniques at reduced computational costs. The approach utilised for matching sequences of images in the above technique has been employed to investigate the improvement in VPR performance and the computational effort required to execute VPR when utilising a sequence-based filtering approach. As CNNs are computationally demanding, this thesis shows that VPR can be performed more efficiently using lightweight techniques. Furthermore, this thesis also investigates the effects of JPEG compression for VPR applications, where important reductions in both transmission and storage requirements can be achieved. As the VPR performance is drastically reduced, especially for high compression ratios, this thesis shows how a fine-tuned

CNN can achieve more consistent VPR performance on highly JPEG compressed data (i.e. above 90% JPEG compression). Sequence-based filtering is introduced to overcome the performance loss due to JPEG compression. This thesis shows that the size of a JPEG compressed image is often smaller than the size of the image descriptor, and therefore should be transferred instead. Furthermore, our experiments also show that the amount of data required for transfer is reduced with an increase in JPEG compression, even when requiring an increased number of images in a sequence. This thesis also analyses the effects of image resolution on the performance of handcrafted techniques, to enable efficient deployment of VPR solutions on commercial products. The analysis performed in this thesis confirms that local feature descriptors are unable to operate on low-resolution images, as no keypoints (salient information) are detected. Moreover, this thesis also shows that the time required to perform VPR is reduced with a decrease in image resolution.

Contents

List of Figures

List of Tables

Abbreviations

AUC Area-Under-the-Precision-Recall-Curve

BNNs Binary Neural Networks

BoW Bag-of-Words

BRIEF Binary Robust Independent Elementary Features

CenSurE Center Surround Extremas

CNN Convolutional Neural Network

DCT Discrete Cosine Transform

DTW Dynamic Time Warping

FAB-MAP Frequent Appearance Based Mapping

GPS Global Positioning System

GPU Graphics Processing Unit

HMM Hidden Markov Model

HOG Histogram-of-Oriented-Gradients

JPEG Joint Photographic Experts Group

LM-DTW Local Matching Dynamic Time Warping

MCN Minicolumn Network

PCA Principal Component Analysis

PCU Performance-per-Compute-Unit

RNN Recurrent Neural Network

ROIs Regions of Interest

SIFT Scale-Invariant Feature Transform

SLAM Simultaneous Localisation and Mapping

SURF Speeded-Up Robust Features

VLAD Vector of Locally Aggregated Descriptors

VPR Visual Place Recognition

WI-SURF Whole-Image SURF

Chapter 1

Introduction

During the last few decades, the rapid advancements in computing power and an ever growing desire to create autonomous platforms resulted in new technologies to emerge. Hence, self-driving vehicles and autonomous robots are no longer a figment of imagination, being widely employed to solve diverse problems ranging from transportation to healthcare and agriculture. Despite that extensive research is still required to achieve robust robot perception to support long-term autonomy, autonomous robots are here to stay.

1.1 Background

Over the last few decades, the robotics community has been dedicated to developing robots that are capable of performing a variety of activities autonomously and reliably. One such example of robot autonomy are self-driving cars [1], where research in this area has been conducted since the middle of 1980s by not only universities and research centers, but also car companies. To advance the development of self-driving cars, the Defense Advanced Research Projects Agency (DARPA) organised a series of competitions in 2004, 2005 and 2007 respectively. The first competition, entitled DARPA Grand Challenge, took place over 142 miles of desert trails in the Mojave Desert and required the self-driving cars to complete the course in under 10 hours. However, there was not a single vehicle that was capable of completing this challenging task. Therefore, the DARPA Grand Challenge was repeated the following year [2], with the Stanford University's car entitled Stanley [3] claiming the fist place when it successfully navigated a 132 miles long challenging environment. The final competition, entitled DARPA Urban Challenge [4], required self-driving vehicles to successfully operate in a 60 miles long simulated urban environment, and completing the task in 6 hours or less.

DARPA Grand Challenge, 2005: Stanley

DARPA Urban Challenge, 2007: Boss

Figure 1.1: The winners of the two DARPA competitions.

The first place in this competition was awarded to Boss [5], Carnegie Mellon University's car. To navigate through this challenging environments, all cars were equipped with a variety of sensors ranging from Light Detection and Ranging (LiDAR), odometer, camera and ultrasonic sensors to Global Positioning System (GPS). The winners of the two competitions are shown in Fig. 1.1.

To achieve robot autonomy, a robot is required to function and take decisions free from on-going human supervision. Therefore, they can operate in constantly changing environments, as their behaviour can adapt to new conditions. To become autonomous, a robotic platform is required to perceive, plan and execute the given task. An important challenge within the area of perception is the localisation process (discussed in sub-section 1.2.1) which is performed utilising the visual information taken from the on-board camera. Moreover, visual localisation can be performed in GPS restricted environments such as urban environments, where the signal is unable to reach areas with large buildings and tunnels [1].

Visual Place Recognition (VPR) is an integral part of computer vision, whose goal is to determine through visual inputs whether an autonomous robot is in a previously visited location. In addition to the changes in the robot's camera pose, our ever-changing environments greatly contribute to the drastic increase in the difficulty of correctly performing place matching, as depicted in Fig. 1.2. Due to the absence of distinct features that are found in confusing and feature-less images, a geographically-different but visually-similar place may not be correctly determined by a VPR system (perceptual-aliasing) [6, 7]. Moreover, as robotic platforms are required to perform place matching in real-time, the storage and processing power limitations must be taken into consideration as they can hamper with the VPR process.

Prior to the deployment of deep-learning in VPR systems, handcrafted local descriptors

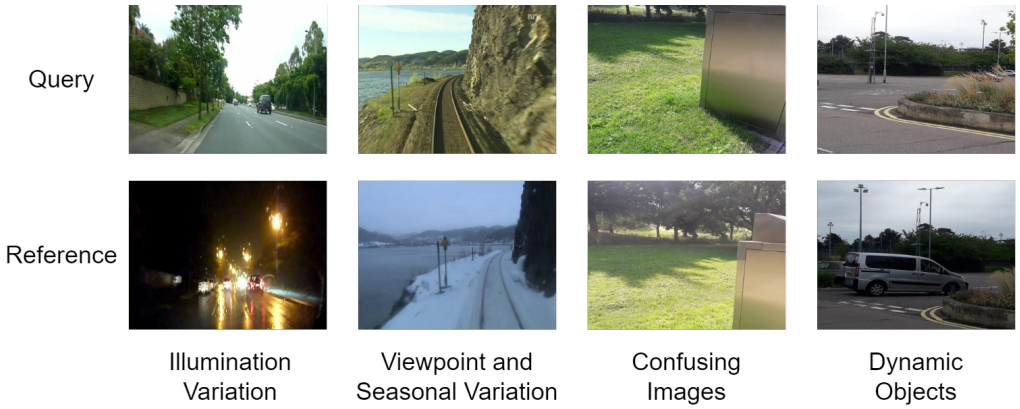


Figure 1.2: Images showing the same place under different conditions.

were used to perform place recognition as discussed in more detail in chapter 2. Local feature descriptors only process salient parts (keypoints) of the image, however these cannot handle severe illumination changes in the environment. In contrast with the local feature descriptors, global feature descriptors process the entire image regardless of its content, but cannot handle viewpoint variation [8]. The application of deep-learning, especially CNNs, was first studied by Chen *et al.* in [9] and since then most of the advances in VPR have been primarily due to deep-learning-based techniques. CNNs are systems capable of learning features extracted from images using supervised training on labeled datasets. Such CNN-based VPR techniques have achieved state-of-the-art performance on the most challenging VPR datasets, as evaluated in [10] and [11]. However, in order to train a CNN for VPR tasks, one needs a large-scale dataset of labeled images taken from different environments, under various angles, seasons and illumination conditions. Although labelled VPR datasets exist, such as Oxford Robot Car dataset [12], SPED dataset [13] and Pittsburgh dataset [14], they represent a particular environment under limited conditional (environmental changes such as seasonal and illumination variation) and viewpoint changes. Therefore, the creation of a large-scale, labeled dataset representing all the different possible variations is not feasible and requires significant time and resources. Furthermore, training a CNN within reasonable times to adjust to a new environment will require dedicated Graphics Processing Units (GPUs) and may take several days/weeks in order to be trained. However, once a CNN has been successfully trained for VPR, the time required to perform place matching is considerably reduced when compared with the time required for training. Even with this significant reduction in time between training and running a CNN-based VPR technique, the matching time and memory footprint of CNNs are significantly higher than those needed for the

handcrafted feature descriptors [10]. Although the CNN-based VPR techniques have largely outperformed handcrafted feature descriptor-based techniques on the image matching front, their intense computational requirements make them harder to use in this field. As a result of these demanding requirements, the deployment of CNN-based techniques for VPR are restricted for resource-constrained vehicles such as battery-powered aerial, micro-aerial and ground vehicles, as discussed in [11] and [15].

To autonomously operate in an environment, a mobile robot has to map, localise and navigate through the environment. This problem of simultaneously mapping and localising the environment is a widely researched topic within the autonomous robotics field, termed as Simultaneous Localisation and Mapping (SLAM) [16]. Generally, robots are equipped with a wide variety of sensors such as cameras, lasers and wheel encoders, that provide essential information that enables motion and location estimates. However, iterative location estimates based on dead-reckoning accumulate errors, which become significant over longer trajectories, leading to incorrect belief about the robot's location in the world. Within autonomous robotics, these accumulated errors can be catered-for if the robot revisits and recognises a previously visited (known) place in the world – action generally labelled as 'loop-closure'. However, it is of great importance that erroneous loop closure detection is avoided [17] as it can obstruct with the SLAM framework. For a vision-only system, this loop-closure can be achieved if a robot is able to recall a previously visited place using only visual information. This task is performed by extracting distinct features from images, calculating their similarities and determining the confidence metrics [17]. If the visual information taken from the camera does not correspond with any location that has already been visited, this new observation is included in the robot's map. Hence, the ability of correctly recognising a previously visited place has become a subject of great interest within the robotic vision community and therefore VPR has developed as a dedicated field within autonomous robotics over the past 17 years [8].

1.2 Simultaneous Localisation and Mapping (SLAM)

Simultaneous Localisation and Mapping (SLAM) [16], one of the most famous research topics, is the process of building a map of an unknown environment while estimating the robot's position relative to the map. This is a challenging problem as the robot is constantly required to correctly match incoming sensory information (query images) with the information

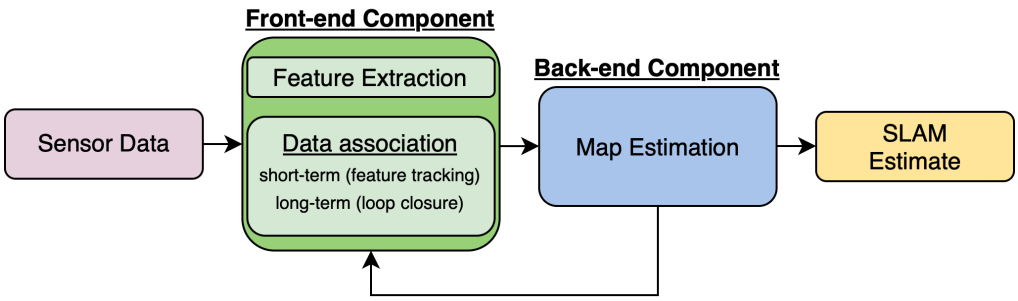


Figure 1.3: The two main components of a SLAM system.

that is stored in the database (the map of the environment), for consistent map generation. A SLAM pipeline typically consists of a front-end component, responsible for handling the sensor data, whilst the back-end component determines the location based on the information received from the sensors [17]. More specifically, the front-end component is responsible for analysing the data retrieved from the camera, and extracting the salient features within the image. Moreover, the front-end is also required to perform data association, where each measurement is associated to a specific landmark (e.g. 3D point) [16]. Loop-closure is performed by the front-end component, where a constraint is added each time the robot returns to a previously known place [18]. Furthermore, erroneous loop closure detection (false-positive loop-closure) is handled by the back-end component [19, 20, 21]. The architecture of a SLAM system including the font-end and back-end components are presented in Fig. 1.3. The back-end component of a SLAM system can provide feedback to the front-end component to determine if a loop-closure has been found.

1.2.1 Localisation

Localisation is the process of determining the robot's location in the stored map, utilising visual information. In contrast with localisation, the pose of a robot refers to its orientation in the environment. The localisation process of a SLAM system is performed by comparing the visual data taken from the camera with the stored map of the environment, to determine the exact location of the robot. However, this process is rendered difficult due to the extreme changes in the appearance of an environment, as shown in Fig. 1.2. Global localisation, also termed as the 'wake-up robot problem' [22], refers to the difficulty in determining the robot's location, given a map of the environment. The re-localisation task (or the 'kidnapped-robot problem' [23]), tries to retrieve the robot's location in the environment following an arbitrary

change in its position, under heavy occlusions or tracking failures [17]. In contrast with global localisation where no previous location information is available to the robot, in local localisation prior knowledge regarding the location of the robot within the map is required.

1.2.2 Mapping

Mapping the trajectory of an autonomous robot during the environment exploration plays a key role in the SLAM architecture as it allows for successful localisation and navigation. As the robot constantly explores a large number of environments, the storage requirements are drastically increased [8]. Moreover, the map needs to be constantly updated as the robot explores the terrain. Pure image retrieval only stores visual information regarding each visited place, with no associated pose information. These maps assume that place matching is only performed on the similarity between the places utilising image retrieval techniques [24]. Metric maps, such as the 2D occupancy grid, only incorporate metric information for high levels of localisation accuracy. Each cell in the occupancy grid map represents a single unit of space, either representing a free space or an obstacle that the robot has to avoid. These maps enable centimeter-level localisation precision whilst also giving the robot geometrically accurate depictions of its environment [25]. Topological maps are a graph-based representation of the environment that include the relative position of places, without storing the metric information [26, 27]. Topological information can facilitate an increased number of correctly determined place matches, while reducing the risk of obtaining an incorrect place match [28]. Topological-metric (hybrid) maps contain both metric (such as distance and direction) and topological information. In hybrid maps, a graph-based model is employed to represent the environment, and the nodes of the model are related to local metric maps [29, 30, 31, 32].

1.3 Visual Place Recognition Overview

VPR is usually cast as an image retrieval problem. The objective of a VPR system is to match images of a place under varying viewpoint and environmental conditions, such as appearance and illumination changes, as shown in Fig. 1.2. To achieve this challenging task, a VPR system searches the best representation of a query image (e.g. a frame taken from a robot's camera while exploring its environment), in the stored map of the environment (e.g. a previous traverse of the same route). The feature descriptors of both query and map images are computed using handcrafted or deep-learning-based approaches, as later discussed in chapter

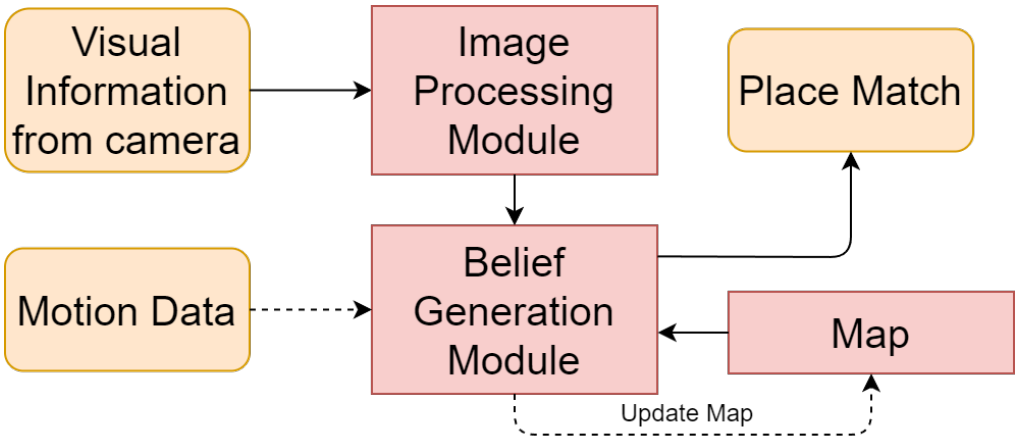


Figure 1.4: The three main components of a VPR system are presented here.

2. Thus, given the feature descriptor of a query image, a VPR technique is required to find the most representative reference descriptor, and hence, the matching place.

Each Visual Place Recognition system has three main components [8]. The first component, namely the image processing module is concerned with the processing of incoming visual data. Additionally, the robot uses a map to store the world's representation of a visited place. The third component is the belief generating module, which is involved in comparing the visual data with the map in order to decide the position of the robot in the environment. These components are shown in Fig. 1.4.

1.4 Thesis Contributions

The contributions of this thesis to the Visual Place Recognition community are as follows:

1. The first contribution presented in this thesis is a dynamic sequence-based and training-less VPR technique for changing environments. In contrast with VPR techniques that utilise a constant sequence length, our technique can successfully adapt its sequence length depending on the environment. This is achieved by analysing consecutive query images to determine a minimum sequence length. Entropy computation of salient regions extraction is then utilised to formulate a dynamic sequence length, that is capable of adapting to distinct environments. Hence, our technique can determine the most representative sequence length for each query image, for achieving the best place matching performance in changing environments.

2. The second contribution of this thesis is an in-depth analysis of the sequence-based filtering schema presented above. As this matching schema is agnostic to the underlying single-frame-based VPR technique, this enabled a comparison between the place matching performance and computational efficiency of each VPR technique tested. A detailed comparison between the single-based image matching schema and sequence-based filtering matching schema is provided. For every VPR technique, the single-frame matching performance is compared with the sequence matching performance. Moreover, the Performance-per-Compute-Unit (PCU) and the boost in performance resulted by introducing sequence-based filtering are assessed.

3. The third contribution of this thesis is a study of the effects of Joint Photographic Experts Group (JPEG) compression in VPR. We show that JPEG compression can be utilised to reduce the amount of data transmitted in decentralised VPR applications. The analysis performed on several JPEG compressed datasets, shows that every VPR technique employed has a drastic decrease in place matching performance, especially in the higher spectrum of JPEG compression. We also show how fine-tuning a CNN specifically for JPEG compressed imagery can enable more consistent and accurate place matching performance.

4. The fourth contribution of this thesis overcomes the loss in VPR performance when utilising highly JPEG compressed images by introducing sequence-based filtering. An in-depth analysis is provided that details the sequence length required to achieve maximum place matching performance, the amount of data transferred on each dataset throughout the entire spectrum of JPEG compression as well as the VPR performance of each technique when the query and reference images have different levels of compression applied.

5. The fifth contribution of this thesis provides an in-depth analysis on the effects of image resolution on the performance of several well-established handcrafted VPR techniques. We confirm that local feature descriptors are unable to operate on small resolution images. We utilise the total time required to perform VPR as a measurement of computational efficiency, showing how a reduced image resolution enables a more efficient VPR process. Moreover, a trade-off analysis between performance and computation is presented, to allow efficient deployment of VPR solutions on low-end commercial products that have limited computational ability, such as battery-powered aerial and

micro-aerial, as discussed in [11, 15, 33].

1.5 Thesis Structure

The remainder of this thesis is organised as follows:

Chapter 2 presents an overview of the literature for Visual Place Recognition. Handcrafted-feature descriptors and deep-learning-based VPR techniques utilised for VPR applications are presented. An overview of the literature regarding sequence-based VPR techniques is presented alongside with details related to the datasets and the performance metrics employed for VPR evaluation.

Chapter 3 presents a new handcrafted VPR technique entitled ConvSequential-SLAM, based on Histogram-of-Oriented-Gradients (HOG) descriptors that is successfully able to perform VPR using an adaptive sequence-based matching approach to tackle VPR in dynamic environments. The proposed technique achieves comparable place matching performance with state-of-the-art VPR techniques on viewpoint and appearance variant datasets.

In chapter 4, the matching schema proposed in ConvSequential-SLAM has been employed to investigate the application of sequence-based filtering on top of single-frame-based methods. In particular, the thesis analyses the VPR performance improvement and the computational effort required to execute VPR using a sequence of images compared with the single-frame approach. The trade-off between VPR accuracy and computational efficiency is also examined, showing how lightweight techniques can replace state-of-the-art descriptors to perform VPR more efficiently, without any loss in place matching performance.

In chapter 5, the thesis investigates the effects of JPEG compression for decentralised VPR applications, where it can be employed to drastically reduce the amount of data transmitted over a communication channel as well as the size of the dataset. An assessment of several well-established VPR techniques under mild to extreme JPEG compression rates is performed on datasets designed for VPR applications. This thesis demonstrates how a fine-tuned CNN-based descriptor on highly JPEG compressed data can achieve higher and more consistent VPR performance than non-optimised VPR techniques. The experiments conducted show that our model is more consistent on both uniform and non-uniform JPEG compressed data than any other technique tested.

In chapter 6, the thesis incorporates sequence-based filtering in a number of well established, learnt and non-learnt VPR techniques to overcome the performance loss resulted

by introducing high levels of JPEG compression. The sequence length that enables 100% place matching performance is reported and an analysis of the amount of data required for each VPR technique to perform the transfer on the entire spectrum of JPEG compression is provided. Moreover, the time required by each VPR technique to perform place matching is investigated. The results show that it is beneficial to use a highly compressed JPEG dataset with an increased sequence length, as similar levels of VPR performance are reported at a significantly reduced bandwidth. The results presented in this chapter also emphasize that there is a trade-off between the amount of data transferred and the total time required to perform VPR. Our experiments also suggest that it is often favourable to compress the query images to the same quality of the map, as more efficient place matching can be performed.

In chapter 7, the thesis investigates the effects of image resolution on the accuracy and robustness of well-established handcrafted VPR pipelines. An assessment of the place matching performance of several handcrafted VPR techniques on various image resolutions is presented. This chapter also reports the total time required to perform VPR for various image resolutions and presents a trade-off analysis between performance and computation.

Chapter 8 presents the conclusions, where a summary of the novel work undertaken in this thesis is presented. The future research directions are also highlighted.

1.6 List of Publications

1. **M. -A. Tomiță**, M. Zaffar, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Convsequential-slam: A sequence-based, training-less visual place recognition technique for changing environments," IEEE Access, vol. 9, pp. 118673–118683, 2021, doi: 10.1109/ACCESS.2021.3107778. [34]

2. **M. -A. Tomiță**, M. Zaffar, B. Ferrarini, M. J. Milford, K. D. McDonald-Maier and S. Ehsan, "Sequence-Based Filtering for Visual Route-Based Navigation: Analyzing the Benefits, Trade-Offs and Design Choices," in IEEE Access, vol. 10, pp. 81974-81987, 2022, doi: 10.1109/ACCESS.2022.3196389. [35]

3. **M. -A Tomiță**, B. Ferrarini, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Visual Place Recognition with Low-Resolution Images," in IEEE ICRA 2023 Workshop on Active Methods in Autonomous Navigation. [36]

4. **M. -A Tomiță**, B. Ferrarini, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Data

Efficient Visual Place Recognition Using Extremely JPEG-Compressed Images", *Under-review*. [37]

5. **M. -A Tomiță**, B. Ferrarini, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Data-Efficient Sequence-Based Visual Place Recognition with Highly Compressed JPEG Images", *Under-review*. [38]

Chapter 2

Literature Review

This chapter presents an in-depth overview of the literature in the field of Visual Place Recognition (VPR). Environmental changes such as illumination [39] and viewpoint variation [6] make a place appear differently on different traverses. These appearance changes render VPR a challenging problem motivating significant effort put by the research community in proposing improvements to existing VPR methods and new techniques. A thorough review of existing research, current challenges and the application of VPR are presented by Lowry *et al.* in [8], Zeng *et al.* in [40], Masone *et al.* in [41] and more recently by Tsintotas *et al.* in [17]. Garg *et al.* [42] present a detailed comparison between VPR in the fields of computer vision and robotics, respectively.

2.1 Local Feature Descriptors

Local feature descriptors such as Scale-Invariant Feature Transform (SIFT) [43] and Speeded-Up Robust Features (SURF) [44] make use of the most notable features in the image for extraction (keypoints), followed by description. This can be seen in Fig. 2.1 a) (image taken from [8]), where the circles represent the salient parts of the image as extracted by SURF. These local descriptors have been widely used to perform VPR such as in [45, 46, 47, 48, 49]. Frequent Appearance Based Mapping (FAB-MAP) [50] is an appearance based place recognition system based on local feature descriptors integrated within a SLAM system. It represents visual places as words and uses SURF for feature detection. The system is successfully able to deal with perceptual aliased images and can perform loop-closure detection. CAT-SLAM [51], extends the work of FAB-MAP by including odometry information. Binary Robust Independent Elementary Features (BRIEF) [52] holds comparable recognition accuracy with

a) b)

Figure 2.1: The difference between a) local feature descriptors and b) global feature descriptors is shown here.

both SIFT and SURF but at reduced encoding times. The authors of ORB [53] propose a computationally-efficient descriptor, capable of performing real-time VPR. The proposed technique is tolerant to noise and is rotation invariant, however it is not scale invariant. Center Surround Extremas (CenSurE) [54] introduces a suite of new feature detectors that outperform the previously mentioned local feature descriptors, performing real-time detection and matching of image features. CenSurE has been used by FrameSLAM in [55]. Bag-of-Words (BoW) [56] and Vector of Locally Aggregated Descriptors (VLAD) [57] build an image descriptor of fixed length by aggregating local feature descriptors around centroids. They are used to partition the feature space in a fixed number of visual words, that enables more efficient image matching. BoW has been used for VPR in [57, 58] and VLAD in [33].

2.2 Global Feature Descriptors

In contrast with local feature descriptors which extract interesting parts of the image, whole-image descriptors process the entire image regardless of its content, as seen in Fig. 2.1 b). Such a whole-image descriptor is GIST [59, 60], that processes the entire image without looking for keypoints in the image. This is done by dividing the image into grids and then processing each separate block. The work done in [61, 62] and [63] shows some examples of GIST whole-image descriptor used in place recognition. Badino *et al.* [64] proposed a variation of SURF, named Whole-Image SURF (WI-SURF), that integrates the accuracy resulting from metric methods together with the robustness of topological localisation, to perform visual localisation. HOG [65] is a global descriptor used to represent gradient angles, whilst also indicating the gradient magnitude for all image pixels. HOG is computationally efficient and tolerant to appearance changes [7]. McManus *et al.* used HOG for VPR in [66].

More recently in CoHOG [67], the authors proposed a training-free technique based on the HOG descriptor that is able to achieve state-of-the-art performance in changing conditions. The proposed approach has zero training requirements and low encoding times, hence it is a great alternative to more resource-intensive VPR techniques, especially for deployment on resource constrained robotic platforms.

2.3 Complementarity of Visual Place Recognition Techniques

As shown in several comparison works [10, 68], there is no universal VPR descriptor that can handle every environmental change. Thus, it is often the case that VPR descriptors work well with some place changes while not with others. The use of complementary VPR techniques is an emerging approach to address VPR. In an attempt to overcome these limitations, the work presented in [69] examines the strengths and weakness of various VPR approaches and optimal combinations of methods are proposed for different environmental conditions. SwitchHit [70] relies on complementarity to propose a switching system to select the optimal VPR technique in dynamic environments.

2.4 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are known to be robust feature extractors and their performance on VPR related tasks showed promising results, thus being extensively explored in the field of VPR in changing environments. The applicability of deep-learning in VPR has been originally studied by Chen *et al.* in [9] where the authors combined all 21 layers of the Overfeat network [71] trained on ImageNet 2012 dataset together with the spatial and sequential filter of SeqSLAM [72]. In [13], the authors trained two neural-network based VPR techniques. The first architecture, entitled HybridNet, uses weights learnt from the top 5 convolutional layers of CaffeNet [73], while the second architecture, AmosNet, was trained from scratch on the Specific PlacEs Dataset (SPED). Arandjelović *et al.* [74] introduced a new layer based on a generalised VLAD entitled NetVLAD, that can be incorporated in any CNN architecture for VPR training. The authors of [10] tested the performance of NetVLAD on multiple datasets, including: Berlin Kudamm [75], Gardens Point [39, 76] and Nordland [77, 78] datasets, showing its robust performance given various VPR scenarios. Cross-Region-Bow [79] achieves viewpoint tolerance by building an image representation from a pre-trained CNN. First, it searches for local maxima in a pre-trained CNN's feature map to identify Re-

gions of Interest (ROIs). Then, the features underlying the selected ROIs are pooled to form an image descriptor using BoW. Khaliq *et al.* [80] present a light-weight CNN-based VPR system, with low memory and resource utilisation, that is robust to viewpoint and environmental changes. CALC [81] trained a Convolutional Auto-Encoder to output illumination-invariant HOG descriptors, where instead of using the original version of the image, laterally shifted and distorted versions of the image are used as input to output the same HOG descriptor for all distorted inputs. This results in a light-weight system robust to variations in viewpoint and illumination. However, CALC has low accuracy when compared to other CNN-based VPR techniques. Torii *et al.* proposed in [82] a place recognition approach, entitled DenseVLAD, that successfully combines synthesis of novel virtual views with a densely sampled but compact image descriptor. Khaliq *et al.* present RegionVLAD [83], a light-weight CNN-based VPR technique that is able to detect salient features from images, while filtering out confusing elements. While RegionVLAD is based on the same approach as Cross-Region-BoW, it employs VLAD for feature pooling. In [39], the authors have used the AlexNet ConvNet [73] pre-trained on the ImageNet ILSVRC dataset [84] for object recognition. The authors of [85] propose SuperGlue, a CNN that matches local features by finding correspondences between the points from two images, while also running in real time. The authors of PointNetVLAD [86] utilise deep-learning to solve point cloud based retrieval for VPR. The "lazy triplet and quadruplet" loss functions are proposed to solve retrieval tasks. Patch-NetVLAD [87] combines the advantages of local and global feature descriptors to achieve improved performance over NetVLAD. The proposed system is tolerant to viewpoint, illumination and seasonal variations while at the same time it is computationally efficient. In [88], the authors propose a VPR technique entitled DELG, that unifies local and global image features for accurate and efficient image retrieval. The authors of [89] present HF-Net, a hierarchical localisation approach for large scale VPR that is robust and accurate to appearance changes, while performing real-time place matching. The authors of [90] and [91] utilise Binary Neural Networks (BNNs) for VPR. These systems are less computationally demanding than other CNN-based VPR techniques, while achieving similar place matching performance as full-precision systems. However, BNNs require dedicated hardware or an inference engine that enables an efficient computation of bitwise operations. Bio-inspired algorithms are also considered to address efficient VPR. Arcanjo *et al.* [92] proposed a lightweight network inspired by Drosophila neural system consisting in a pre-processing stage to compute a compact binary image representation, followed by a classifier to predict the current location of the robot.

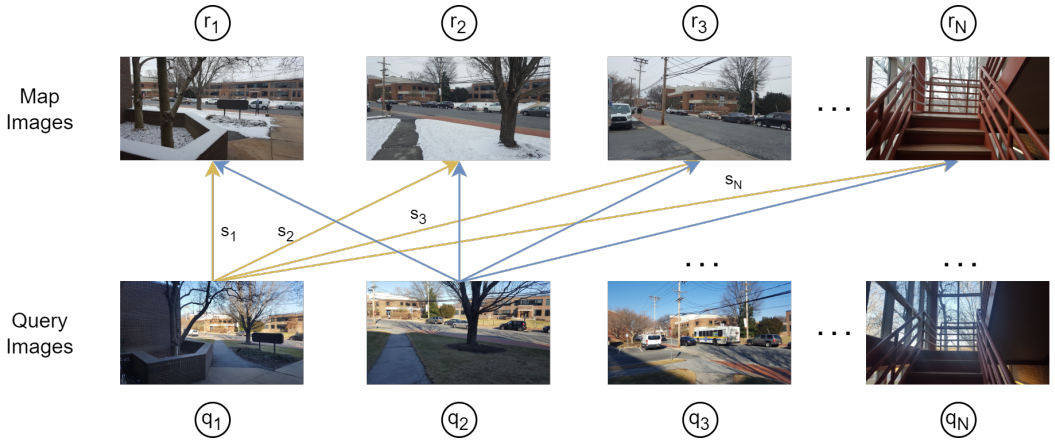


Figure 2.2: The single-frame matching approach is presented here.

2.5 Sequence-based Visual Place Recognition Techniques

The most common approach utilised for analysing sequential information is to perform sequence matching [72, 93]. In comparison with the approaches previously mentioned in sub-sections 2.1, 2.2 and 2.4 that represent each map image as a distinct place, the VPR techniques presented in this sub-section create groups/sequences of images to improve their place matching performance. In the presence of severe seasonal and illumination variations, sequence-based VPR techniques usually yield increased VPR performance over the single-based image descriptors, as later discussed in this sub-section. In the single-frame matching schema, every image q_i captured from a robot's camera (e.g. the query) is matched with every reference image r_i present in the map. A similarity score s_i is computed for every query-reference pair. Hence, the reference image r_i with the highest similarity score s_i is retrieved as the matching place for q_i. For sequence-based VPR techniques, sequences of query and reference images of length **K** (where $\mathbf{K} \geq 2$) are created and matched together. A similarity score is generated between each query-reference sequence. The sequence of reference images $rseq_i$ with the highest similarity score s_i is regarded as the matching place for any given query sequence $qseq_i$. Fig. 2.2 presents the single-frame matching approach, whilst the sequence-based filtering schema is shown in Fig. 2.3.

In SeqSLAM [72], sequences of camera frames are compared instead of single frames, thus achieving increased performance in VPR when compared to traditional feature-based techniques, in scenarios where the place is subject to drastic changes. The comparison algorithm of SeqSLAM finds multiple strong matching candidates within every local section of the route. Within the local best matches, spatially coherent sequences are determined by com-

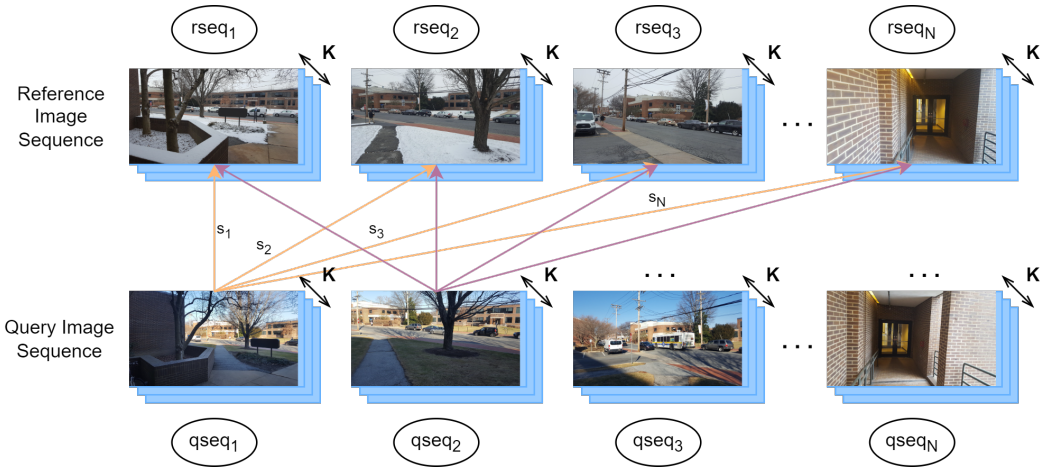


Figure 2.3: The sequence-based matching approach is presented here.

paring each frame to all previously learnt frames to form an image difference matrix. A linear search is then performed through the image difference matrix to find possible localisation hypotheses. The work of Pepperell *et al.* [94] on SMART extended SeqSLAM by incorporating odometry into its calculations. The authors of [95] proposed a new sequence-based VPR system for aerial robots, that uses Bayes estimation to perform sequence image matching. Moreover, it does not require that the query image sequence to be organised in the same order as the stored map. In [96], the authors present a fast and compact VPR pipeline where sequence matching is used to resolve the collisions in the hash space. It has an overall low storage footprint while at the same time having extremely fast retrieval and sub-linear storage growth. Johns *et al.* [97] show a new method for appearance-based localisation, namely Feature Co-occurrence Maps. The performance of this technique does not degrade during severe changes in illumination, thus place matching is performed at high precision/recall. Co-occurrence Maps outperforms both FAB-MAP [50] and SeqSLAM [72]. The authors of DeepSeqSLAM [98] proposed to integrate a Recurrent Neural Network (RNN) model on top of a CNN. The resulting system is successfully able to learn both visual and positional representations from a single monocular image sequence of a route. The authors of [99] propose a sequence-based VPR system with robust localisation, by creating a data association graph that is able to relate images from sequences. This approach resulted in a VPR system than can successfully handle substantial seasonal change. Vysotska *et al.* presented in [100] a new approach based on graph-based image sequence matching that is swiftly able to retrieve the correspondences between a sequence of query and reference images, under severe appearance changes. The authors of [101] use a Minicolumn Network (MCN) approach that is

able to create an internal representation that encodes sequential information. The authors of [102] propose two filtering approaches for sequences of images, the Hidden Markov Model (HMM) and Monte Carlo-based visual localisation. The proposed technique is robust to appearance changes. The authors of [103] present an approach for scene boundary detection, that utilises dynamic segmentation of the visual data instead of a pre-determined sequence length. The authors of [104] propose a lightweight sequence-based loop-closure detection system based on Principal Component Analysis (PCA) that is successfully able to decrease the dimensions of the image descriptor, resulting in reduced computational complexity. Moreover, to reduce the matching time and improve the matching efficiency of image sequences, consecutive sequences of query images are combined with fast approximate nearest neighbor search (ANNS). ANNS provides an approximate solution to the nearest neighbour search, by trading off some levels of accuracy for faster search times [105]. In [106], odometry is combined with a coarse and a fine localisation module, to create a sequence-based localisation pipeline, robust to illumination and seasonal variations. In [107], the authors combine the CNN's features with the temporal information found in a sequence of images to construct a graph-based VPR method, capable of outperforming FAB-MAP. In SeqMatchNet [108], the authors propose a triplet loss function based on sequence matching. The authors of [109] propose a VPR algorithm that matches sequences of query and reference frames. A matrix of low-resolution, contrast-enhanced image similarity values are computed in order to perform sequence matching and a HMM framework is used to find the best sequence alignment. However, the system can only deal with small viewpoint variations. STA-VPR [110] is a sequence-based VPR technique that uses an adaptive Dynamic Time Warping (DTW) algorithm in order to improve its robustness to changes in appearance and viewpoint. Furthermore, to achieve image sequence matching based on temporal alignment, a Local Matching Dynamic Time Warping (LM-DTW) algorithm is used, thus achieving a linear time complexity. Both [109] and [110] are suitable to deal with non-linear changes in velocity, whereas [72] does not perform well with variable velocities. It has been shown in [111] that the computational cost required to perform sequence-based matching grows linearly with the size of the map and the number of images in a sequence. As a result of these limitations, the authors of [111, 112, 113, 114, 115] propose to perform sequence retrieval by utilising sequence descriptors.

2.6 Decentralised Visual Place Recognition

Several robotic applications benefit from deploying multiple units operating in parallel, such as search and rescue missions, where multiple robots can cover the search area easier than a single agent (i.e. a system that operates autonomously). This is also the case for planetary exploration, where the dangerous terrain found on other planets can be simultaneously explored by multiple smaller robots. Moreover, due to the extreme nature of the task, employing multiple agents may be desirable to minimise the risk of failure when areas are too dangerous for only a single robot to explore. Multi-agent collaborative tasks are also of great interest to NASA, as deep space exploration continues. In [116], two MarCO spacecrafts have been utilised to simultaneously explore the terrain of Mars, while relaying the captured data back to Earth. However, when a large number of robots are part of a decentralised system, the bandwidth must accommodate the entire system to facilitate swift data transmission between them. This is because the robots are still required to transmit data even when operating in bandwidth constrained environments [117]. Knowing their position in the operating environment as well as the positions of the other agents is fundamental for mobile robots. As part of the visual simultaneous localisation and mapping (SLAM), visual place recognition (VPR) is an essential task for the localisation process when the environment is unstructured, Global Positioning System (GPS) is unavailable or the visual odometry drifts due to accumulated errors. For applications that involve a single robot, an ideal VPR method is accurate in detecting known places and efficient enough to fit the robot's hardware and battery capability [33]. For applications requiring multiple robotic platforms to collaborate, navigate and map the environment effectively, the visual data gathered must be transmitted remotely between each robot [118, 119, 120]. Hence, the amount of data required to be transmitted must be taken into consideration when working with limited bandwidth available for VPR.

In recent years, considerable effort has been put in creating decentralised VPR architectures. In contrast with centralised architectures where each robot sends the map to a central server that performs the place matching computations [121, 122], in decentralised architectures each robotic platform performs the place matching computations between their own map and that of other robots [123]. Thus, the visual data gathered from each agent has to be shared between each robotic platform. As previously mentioned, this is especially important in applications such as search and rescue, where each agent can cover a distinct part of the environment, resulting in a swifter task completion. In similar scenarios, these systems have to be robust and scalable [124], hence a centralised point of failure or dependence

may be undesirable. To achieve a robust decentralised system, the bandwidth limitations of the communication network need to be overcome [117]. The authors of [125] propose a decentralised system for multi-robot exploration based on thermal images and inertial measurements. The front-end of the pipeline handles the feature tracking and place recognition, whilst the back-end component reduces both the memory and computational cost by utilising a covariance-intersection fusion strategy. The communication pipeline employed is based on VLAD, resulting in reduced bandwidth usage. In [126], a method for multi-robot SLAM based on ranging sensors is presented, where the system can create consistent maps even in scenarios where loop closures cannot be detected. In [118], the descriptor space of NetVLAD is clustered, and each smaller cluster is sent to a robot to perform efficient decentralised VPR. In [119], a data-efficient decentralised visual SLAM system is presented, where the data association scales linearly with the number of robots present in the multi-robot SLAM system. The authors of [120] present a loop detection architecture for performing multi-robot underwater visual SLAM. A common observation in [118, 119, 120] and [123] is that in multi-robot SLAM systems, working with a reduced bandwidth can significantly increase the difficulty in transferring the images between multiple autonomous vehicles. The work proposed in chapter 5 and chapter 6 addresses the data transfer problem using JPEG compression to facilitate VPR applications where the available bandwidth is not capable of transmitting the visual data in an uncompressed form.

2.7 Benchmarking Visual Place Recognition Approaches

Three main components are utilised when assessing the performance of a VPR technique namely datasets, ground-truth information and the performance/evaluation metrics [7]. It is usually the case that each dataset is accompanied by the ground-truth information, which explicitly states the correspondence between the query and reference frames. For any dataset, the ground-truth can be a CSV file, numpy array (an array whose elements must have the same data type [127]), GPS information etc. However, for datasets where the query and reference images have the same name or index, the ground-truth information is not always required. In these cases, for any given query image, the reference image with the same name or index should be retrieved as a matching place. A wide variety of datasets for benchmarking VPR applications exist in the literature, and we discuss these in sub-section 2.7.1. Similarly, several performance metrics that enable an efficient performance assessment of a VPR tech-

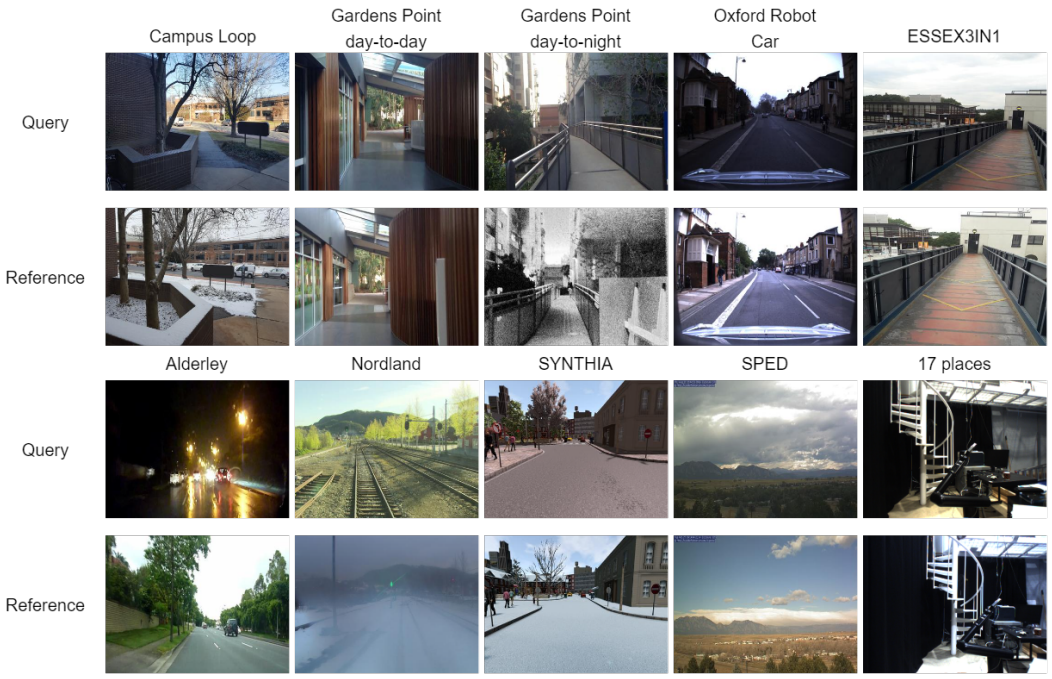


Figure 2.4: Sample images taken from well-established VPR datasets are presented here.

nique are discussed in-depth in sub-section 2.7.2.

2.7.1 Test Datasets for Visual Place Recognition

The datasets utilised are an integral part of the VPR process, as they replicate the environmental changes that a robot can experience during its deployment, and can range from illumination, viewpoint and seasonal variations to confusing features, dynamic objects (such as vehicles) and clutter background. A dataset mainly consists of two traverses which are captured along the same route, during different times and environmental conditions. The first traverse along a route is stored as the map (reference images), while the second traverse represent the query images, which are captured by a robot's camera while exploring its environment.

Several datasets exist that have been widely employed by the VPR community to test the performance of VPR techniques. The majority of studies are performed on publicly accessible datasets, which can include frames taken in various environments, ranging from indoor to outdoor scenes. This thesis utilises a combination of datasets presenting illumination, viewpoint, and weather variations to cover some of the most common viewing conditions experienced by a robot, where the operating environment might present heterogeneous conditions

Table 2.1: A selection of datasets designed for VPR applications.

Dataset	Environment	Variation	
		Viewpoint	Conditional
Nordland [77, 78]	Outdoor	Lateral	Appearance
Alderley [72]	Urban	Lateral	Appearance
Campus Loop [81]	Indoor/Outdoor, Campus	Lateral	Appearance
Gardens Point [39, 76]	Indoor/Outdoor, Campus	Lateral	Illumination
Oxford Robot Car [12]	Urban	Lateral	Illumination
ESSEX3IN1 [6]	Indoor/Outdoor, Campus	3D	Illumination
SYNTHIA [128]	Synthetic Environment, Urban	-	Illumination
17 places [129]	Indoor	Lateral	Illumination
SPED [13]	Outdoor	-	Appearance

in different places. The following datasets have been utilised in various chapters of this thesis and they are as follows: Campus Loop dataset [81] contains sequences of 100 query and 100 reference images taken from both indoor and outdoor locations of a campus environment. The frames are taken under viewpoint and seasonal variations, whilst also containing dynamic objects which contribute towards a challenging dataset for VPR applications. Gardens Point dataset [39, 76] consists of three traverses of the environment, two during the day (*day left* and *day right*) and one traverse captured during the night (*night right*). Thus, the images within the above-mentioned dataset are divided as follows: 200 query images (*day left*) and 400 reference images equally split into day images (*day right*) and night images (*night right*). Apart from changes in the illumination, the dataset also includes viewpoint variation. Nordland dataset [77, 78] captures outdoor images of a train journey in Norway during each season (spring, summer, autumn and winter). Since the most notable differences between seasons are seen during the summer and winter, this thesis uses 172 query and 172 reference images taken from the summer-to-winter traverses of the Nordland dataset. ESSEX3IN1 [6] is a dataset created at the University of Essex, with a focus on perceptual aliased and confusing places. This dataset is composed of 420 frames, equally split into 210 query and 210 reference images. Oxford Robot Car dataset [12] contains 200 query and 200 reference images that are taken under illumination and viewpoint changes. SYNTHIA dataset [128] presents a simulated city-like environment which consists of 200 query and 200 reference

images, whose frames contain various weather, seasonal and illumination changes. The 17 places dataset [129] contains images that are captured in distinct lighting conditions, in multiple indoor environments. For this thesis, three locations have been selected entitled Arena, AshRoom and Corridor. Hence, this dataset consists of 457 query (*day_vme1*) and 434 reference images (*night_vme1*). The Alderley dataset [72] was created in Brisbane, Australia. The first traverse is composed of 201 query images that are captured at night time, during limited environmental illumination and in the presence of rain, translating to low visibility. The second run (201 reference images) was captured during the day. Specific PlacEs Dataset (SPED) dataset [13] contains 607 query and 607 reference frames. These are low-quality images taken from outdoor cameras [130] under weather, seasonal and illumination variations. The images taken from these cameras capture an array of scenes such as forest/mountain landscapes, country roads and urban locations. Sample images taken from each previously mentioned dataset are presented in Fig. 2.4. In Table 2.1, the datasets together with the changes depicted in the environment are included. Other datasets presented in the literature include Berlin A100 [75], Berlin Halenseestrasse [75], Berlin Kudamm [75], Query247 [82], Tokyo24/7 [74], INRIA Holidays [131], Cross-Seasons [132], Corridor [133], Pittsburgh [14] and Living Room [134].

2.7.2 Performance Metrics for Visual Place Recognition

In this sub-section, several evaluation criteria designed for VPR applications are presented.

The authors of [8] suggested that Precision-Recall curves are a key evaluation metric for VPR techniques. Therefore, an ideal system would achieve 100% precision at 100% recall. Area-Under-the-Precision-Recall-Curve (AUC) is widely used in VPR research for evaluation purposes [7] due to the fact that it performs well on unbalanced data, which is also the case for VPR applications. AUC is computed by plotting the Precision-Recall curve at different confidence thresholds as follows:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{2.1}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{2.2}$$

In both equations (2.1) and (2.2), the True Positives (TP) refer to the correctly retrieved matches, False Positives (FP) represent the incorrect retrieved matches while False Negatives

(FN) are the incorrectly discarded matches. The values of AUC are between [0,1], with higher values representing better VPR performance [135]. However, AUC cannot determine whether, for any Recall value, the Precision is at 100% [68].

R_{P100} [136] is a derivative from a PR-Curve and it denotes the Recall value at which the Precision starts to drop from 100%. More specifically, it represents the percentage of TP that can be retrieved with no FP. It has been shown in both [135] and [137] that a FP can have severe consequences, hence R_{P100} is a good performance indicator utilised in evaluation works such as [136] and [138]. However, R_{P100} is not able to determine the performance of a VPR technique in the lower spectrum. For this reason, the authors of [68] introduced a new metric based on R_{P100} entitled Extended Precision (EP) that has the following formula:

$$EP = \frac{P_{R0} + R_{P100}}{2}, \tag{2.3}$$

where, P_{R0} represents the Precision at minimum Recall. The denominator in equation (2.3) is included to ensure that the EP is in range [0,1]. Similarly to AUC, higher EP translates to a better VPR performance.

The authors of [10, 11, 81] and [83] determined that the feature encoding time (t_e) of a VPR system to be an important performance indicator. In [67], the authors evaluated a system's performance using PCU. This is defined by combining precision at 100% recall (P_{R100}) with t_e as in equation (2.4):

$$PCU = P_{R100} \times \log\left(\frac{t_{e\_max}}{t_e} + 9\right) \tag{2.4}$$

In this equation, the maximum feature encoding time (t_{e_max}) is used to represent the most resource intensive VPR technique, while t_e represents the feature encoding times for each of the remaining techniques (where $t_e \leq t_{e\_max}$). It is worth mentioning that without the scalar 9 in equation (2.4), the VPR technique with $t_e = t_{e\_max}$ will always result in a PCU of 0. Techniques with higher precision and lower feature encoding time generally lie towards the higher spectrum of PCU, while compute-intensive and less precise techniques converge towards lower PCU values. Thereby, this addition provides a more interpretable range.

Another metric utilised for VPR evaluation purposes is the accuracy [7], representing the percentage of correctly matched images. This performance metric is computed as in equation (2.5) below:

$$Accuracy = \frac{N_c}{N_q}, \tag{2.5}$$

where N_c represents the number of correctly matched query images and N_q the total number of query images. The accuracy has values in range [0,1]. Higher the accuracy, higher the place matching performance of a VPR technique.

In contrast with the previously mentioned performance metrics that only assess the VPR performance of a technique, the encoding time and matching time are also important especially for VPR applications where resource-constrained platforms are utilised. While the encoding time t_e refers to the amount of time that a VPR technique requires to compute the feature descriptor of an image, the matching time t_m represents the amount of time required to match the descriptor of a query image with all the reference descriptors in the map. Both the encoding time t_e and matching time t_m of various VPR techniques have been discussed at length throughout this thesis.

2.8 Summary

In this chapter, a detailed overview of the literature regarding Visual Place Recognition (VPR) is presented, including handcrafted-feature descriptors and deep-learning-based VPR techniques. An overview of the literature regarding sequence-based VPR techniques as well as an introduction to decentralised VPR have been presented. Moreover, the datasets and performance metrics utilised to assess VPR performance have also been described in this chapter.

Chapter 3

ConvSequential-SLAM: A Sequence-Based VPR Technique

In order to tackle the VPR challenges previously discussed in chapter 1, a large number of handcrafted and deep-learning-based VPR techniques have been developed, where the former suffer from appearance changes and the latter have significant computational needs. In this chapter, a new handcrafted VPR technique is presented, entitled ConvSequential-SLAM, that achieves comparable place matching performance with well-established deep-learning-based VPR techniques in challenging conditions, whilst having a reduced computational footprint which translates to a wider availability for resource constrained platforms. We utilise sequential information and block-normalisation to handle appearance changes, while using regional-convolutional matching to achieve viewpoint invariance. Therefore, our technique employs an adaptive sequence-based matching approach to address VPR in dynamic environments. We analyse content-overlap in between query frames to find a minimum sequence length, while also reusing the image entropy information for environment-based sequence length tuning. The place matching performance of ConvSequential-SLAM is reported in contrast to several well-established VPR techniques on four public datasets.

3.1 Introduction

In this work, we propose a novel sequence-based and training-free VPR technique, namely ConvSequential-SLAM, that is successfully able to perform VPR under changing viewpoint and appearance conditions. In contrast to other sequence-based VPR systems such as [72] and [98] that use a constant sequence length of images, our technique is using a dynamic sequence-based matching approach that is able to determine the most representative sequence length for each sequence of images. The resulting system is a training-less and light-weight VPR system, successfully able to adapt to distinct environments. We report comparable performance with more complex deep-learning-based VPR techniques on both viewpoint and conditionally-variant datasets while having a lower computational load which is important especially for resource constrained platforms.

We make the following main contributions:

- We integrate convolutional matching into our system, achieving robustness to moderate viewpoint variations.
- We achieve conditional invariance by using regional, block-normalised HOG descriptors instead of contrast-enhanced pixel-matching.
- We developed an analysis based on *information-gain* from consecutive query images to determine the minimum sequence length needed. Since ConvSequential-SLAM utilises HOG for descriptor computation and regional convolutional matching for descriptor comparison (refer to sub-section 3.2.4), this approach cannot be directly included in other VPR techniques and it needs to be adapted for each particular technique.
- Building upon the sequence length generated by analysing consecutive query images, we use the entropy computation for salient region extraction to formulate an optimal dynamic sequence length, instead of a constant sequence length, as used in sequence-based VPR techniques.

3.2 Methodology

This section presents the methodology proposed in our work. The query images represent the visual data received from the camera, while the reference images represent the stored map

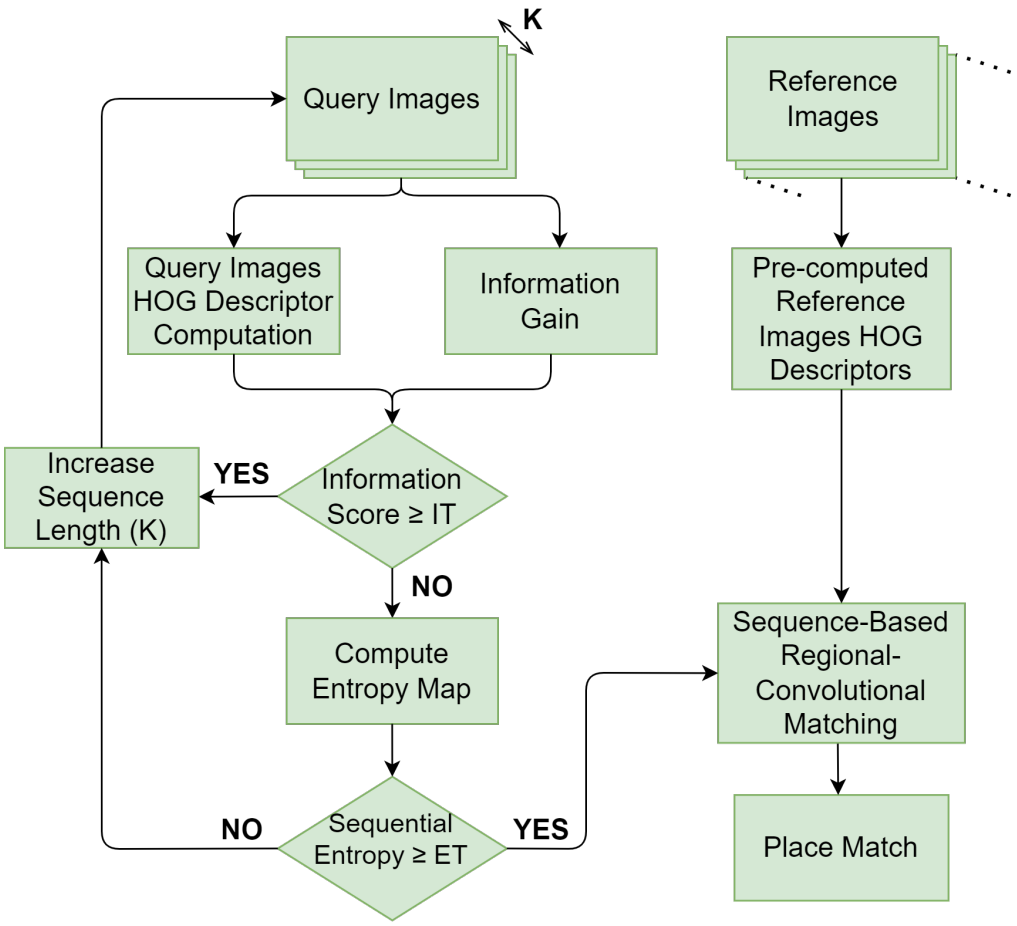


Figure 3.1: The block diagram of our framework is shown here, which presents all the major components of the system.

of the environment in the form of RGB images. The block diagram showing each step of the ConvSequential-SLAM system is presented in Fig. 3.1.

3.2.1 Information Gain

The first major innovation in our work is the ability of our technique to determine the information gain resulted from analysing consecutive query images. This allows a more robust understanding of the environment, while it also gives enough information about different textures and properties found in successive query images. This approach is used to determine the local change-point in consecutive query images, thus enabling a minimum sequence length (min_K) for each sequence of images to be determined (see sub-section 3.2.5).

The information-gain is calculated as follows. Firstly, we compute the HOG of the first and

second query image that are part of a sequence. Secondly, we proceed to compare these two images together using regional convolutional matching (see sub-section 3.2.4), generating a similarity score. Finally, we compare this score with the Information Threshold (IT) to determine if the similarity between the two query images provides sufficient information gain. We then proceed to compare the first query image with the third and so on, repeating the above steps, until we find a representative minimum sequence length. The information gain can be easily summarised as in equation (3.1) and (3.2) below:

$$Information\ Gain = 1 - Similarity\ Score \tag{3.1}$$

$$Initial\ Sequence = \begin{cases} K+1, & \text{if } Information\ Gain \geq IT. \\ entropy\ map, & \text{otherwise.} \end{cases} \tag{3.2}$$

In the above equation, $min\_K \leq K \leq max\_K\_IG$, IT is in range [0,1] and represents the Information Threshold, min_K is the minimum sequence length (set to 1) and max_K_IG is the maximum sequence length. The $Initial\ Sequence$ in equation (3.2) represents the number of query images that are part of the query list generated by this approach. When the Information Gain module provides its best sequence length (e.g. $Information\ Gain < IT$), we proceed to calculate the sequential entropy (see sub-section 3.2.6) for that sequence of query images and determine whether this has the optimal length.

3.2.2 Entropy Map and ROI Extraction

The second step in the ConvSequential-SLAM framework is to create the entropy[1] map representing the salient regions in each query image. The entropy map creation is based on estimating the local pixel intensity variation within the grayscale image and computing the base-2 logarithm of the histogram of pixel intensity values within each local region. This entropy map is represented by a matrix of size $W1 \times H1$, the elements of which are values in the range $\{0-8\}$, due to the pixel intensities being in the range of 2^0 to $2^8 - 1$. The dimensions $W1 \times H1$ represent the fixed size dimensions of the input image. The following matrix represents the entropy map for a query image:

[1] https://scikit-image.org/docs/stable/api/skimage.filters.rank.html#skimage.filters.rank.entropy

$$Entropy = \begin{bmatrix} e_{11} & e_{12} & e_{13} & \dots & e_{1W_1} \\ e_{21} & e_{22} & e_{23} & \dots & e_{2W_1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ e_{H_1 1} & e_{H_1 2} & e_{H_1 3} & \dots & e_{H_1 W_1} \end{bmatrix} \quad (3.3)$$

Where $e_{ij} \in \{0 - 8\}$.

Using the entropy map of an image, we extract Regions-of-Interest (ROIs) by computing the average entropy of a region of size $W2 \times H2$. If this entropy is above a threshold ET, it reflects that a region is informative and is selected as a ROI. The total number of regions (non-overlapping) in an image is $N = W1/W2 \times H1/H2$ and the total number of ROIs is G which can vary from one query image to another. In comparison with the traditional Top-G approach where the value of G always remains constant, our approach offers greater saliency and computational benefits. When dealing with an image that contains numerous non-informative (confusing) regions, this approach only analyses the regions which are determined to contain salient information. Therefore, by utilising a variable number of ROIs instead of the Top-G approach, low-textured images are successfully matched. Moreover, as the confusing regions of the image are discarded before the regional convolutional matching process (refer to sub-section 3.2.4), the computational intensity of ConvSequential-SLAM is reduced. Fig. 3.2 shows the salient regions determined by our technique for various values of ET. It can be seen that by increasing ET, non-informative elements such as walls and floors are filtered out. Thus, with an increase in ET the number of ROIs detected in an image is reduced.

To get a single entropy value for the entire image, all the elements of the entropy matrix are summed, then divided by $W1 \times H1 \times 8$ to get the re-scaled value. This is useful for the computation of sequential entropy of a sequence of query images to determine the dynamic sequence length (see sub-section 3.2.6).

3.2.3 Regional HOG Computation

The process of regional HOG computation takes place as follows. In the first instance, we compute a gradient map of a grayscale image of size $W1 \times H1$. Following this, a histogram of oriented-gradients is computed for all N regions of the image, with each region having the size of $W2 \times H2$. Furthermore, each histogram of every region has L bins, where each bin is labelled with equally spaced gradient angles between 0-180 degrees. Lastly, we use L2-normalisation to achieve illumination invariance [139, 140]. This is computed as the square

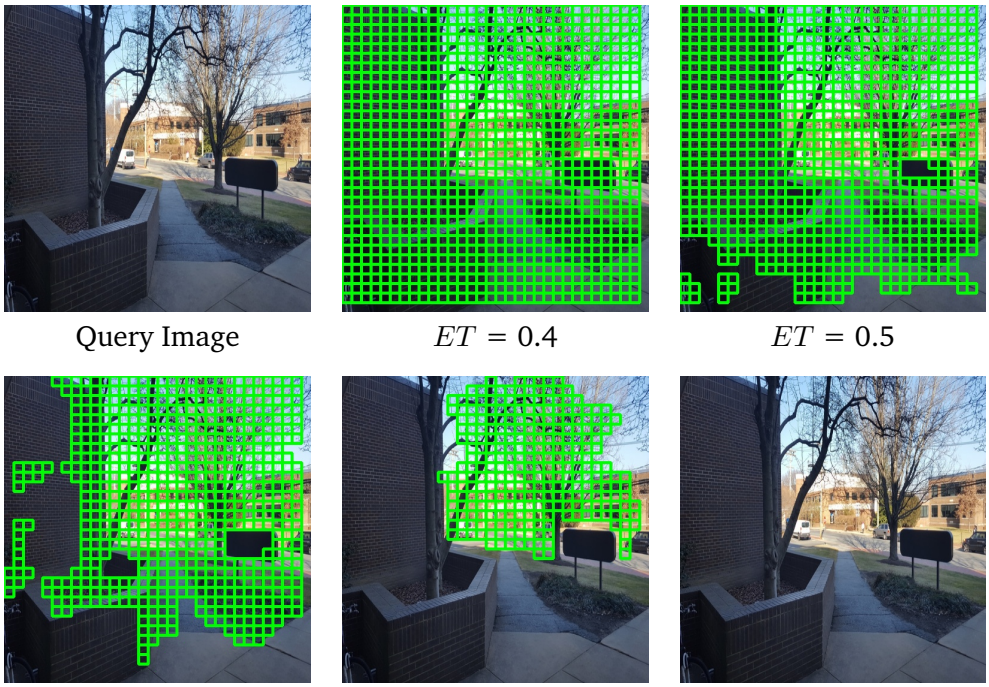


Figure 3.2: The ROIs extracted by ConvSequential-SLAM for various ET.

root of the sum of the squared pixel values. By utilising L2-normalisation, the pixel values are scaled down such that the impact of illumination on the image is reduced. This is done at a block level of size $(W2 \times 2) \times (H2 \times 2)$.

3.2.4 Regional Convolutional Matching

Following the regional HOG computation, we proceed to regional convolutional matching (presented in Algorithm 1), given each query image is represented as N regions, each being described by a HOG-descriptor of depth $4 \times L$. Using the information from the ROI evaluation, these N regions are reduced to G salient regions. By doing so, the query image HOG-descriptor can be represented as a 2D matrix of dimensions $[G, 4 \times L]$. The reference image has N regions with the descriptor size of $4 \times L$, therefore its resulting matrix has the dimensions of $[N, 4 \times L]$. We then proceed to multiply the query and reference matrices ($d1$ and $d2$, respectively), and the result is a matrix (entitled $d1d2dot\_matrix$) of dimensions $[G, N]$. Each row of this matrix represents a salient region of a query image, while each column represents the cosine-matching scores for that region with all the N regions of a reference image. Max-pooling ($d1d2matches\_maxpooled$ in Algorithm 1) is used across the rows of the aforementioned matrix in order to determine the best matched regions between the query

Algorithm 1: The regional convolutional matching process is detailed here.

Given: Query Image HOG-Descriptor ($d1$)

Given: Reference Image HOG-Descriptor ($d2$)

Given: Query Image ROIs ($regional\_G$)

nr_of_regions=$(W1/W2-1) \times (H1/H2-1)$

INITIALISE (2D array of 0s): d1d2dot_matrix[nr_of_regions, nr_of_regions])

INITIALISE (array of 0s): d1d2matches_maxpooled[nr_of_regions]

INITIALISE (array of 0s): d1d2matches_regionallyweighted[nr_of_regions]

```
np.dot(d1, d2, out = d1d2dot_matrix)
// Select best matched reference region for each query region
np.max(d1d2dot_matrix, axis = 1, out = d1d2matches_maxpooled)
// Weighting regional matches with regional_G
np.multiply(d1d2matches_maxpooled, regional_G,
                                    out = d1d2matches_regionallyweighted)
// Compute final score
score = np.sum(d1d2matches_regionallyweighted) / np.sum(regional_G)
```

and reference images. The final score is computed as the arithmetic mean of matching scores of all G regions and is in the range of $0-1$, such that the higher the score, the higher the similarity between the two images. Finally, the reference image that has the highest score is chosen to be the best match for a given query image. The value of the parameters utilised in Algorithm 1 are detailed in sub-section 3.3.3.

3.2.5 Creating the Query Images Sequence

Query images are added into a 1D list in a sequential manner, such that the length of this list is dependent on the sequential entropy (explained in sub-section 3.2.6). Even if the sequential entropy's value for the first **K** images is higher than the Entropy Threshold (ET), where $0 \leq ET \leq 1$, the minimum sequence length will be determined using the information-gain resulted from analysing consecutive query images (see sub-section 3.2.1). Thus, we will not end up with non-optimal sequence lengths, that will ultimately result in poor performance. The 1D query list containing a sequence of query images is represented as:

$$Sequential\ Query\ List = \begin{bmatrix} q_1 & q_2 & q_3 & \dots & q_K \end{bmatrix} \tag{3.4}$$

In the above equation, q_1 is the first query image, q_K is the last query image, and **K** is the total number of images that are part of a sequence.

As previously mentioned, the length of this list will constantly change, but all the images will be in a sequential order, starting from the first image to the **K**-th image. When computing the second sequence of query images, we start with the second image (q_2) and so on. It is important to note that for any N images read, the number of query images sequence lists created will be $N-$ **K** + 1, where **K** will contain the length of the last list created. That is, for any N query images, the algorithm will only match the first $N-$ **K** + 1 images.

3.2.6 Entropy-Based Dynamic Query Images Sequence

The second key innovation is incorporating the ability of our technique to reuse entropy as measure of the overall information content found in a sequence of query images, to decide an optimal sequence length of each query list. Building upon the sequence length generated by analysing consecutive query images (see sub-section 3.2.1), we use the entropy to maximize the efficiency of this length. To achieve this, our technique first looks at the information content (entropy score) of the query sequence list generated in sub-section 3.2.1. If the information content within this sequence of images is less than a threshold (ET), we increase the sequence length by a constant step, then recompute the information content for this new increased sequence of images. If the information content ($Sequential\ Entropy$) for this increased sequence of images reaches a reasonable value (ET), the corresponding length of the query images sequence is used, otherwise we keep increasing it (up to the maximum sequence length) to find a suitable sequence length. $Sequential\ Entropy$ represents the arithmetic mean of the entropy scores e_i of the query images within the sequence. Thus, for any **K** query images in a sequence, the sequential entropy is calculated utilising equation (3.5). The entire iterative process is summarised in equation (3.6).

$$Sequential\ Entropy = \frac{\sum_{i=1}^{K} e_i}{K} \tag{3.5}$$

$$Sequence\ Length = \begin{cases} min\_K\_IG, & \text{if } Sequential\ Entropy \geq ET. \\ K+1, & \text{otherwise.} \end{cases} \tag{3.6}$$

Algorithm 2: Matching Query and Reference Sequences

```
Given: Query Images Sequence (Q_Seq)
Given: Reference_Images_List (R_List)
ref_matching_scores = [ ]
iterator = 0
K = Length (Q_Seq)
while itr + K ≤ Length(R_List) do
    Sequential_Reference_List = R_Seq = [ ]
    for ref_itr in range(itr, itr + K) do
        APPEND R_List[ref_itr] to R_Seq
    match_score = Sequence_Matching_Func(Q_Seq, R_Seq)
    ADD match_score to ref_matching_scores
    iterator = iterator + 1
Best Match = Max (ref_matching_scores)
```

In equation (3.6), $min\_K\_IG \leq K \leq max\_K$, ET represents the Entropy Threshold, min_K_IG is the minimum sequence length (generated in sub-section 3.2.1) and max_K is the maximum sequence length. The $Sequence\ Length$ in equation (3.6) represents the number of images that are part of the query list at a given time, thus being dependent on the value of **K**. In the same equation, the $Sequential\ Entropy$ refers to the average entropy value (see sub-section 3.2.2) of **K** images that are part of this query list.

3.2.7 Dynamic Sequence Matching

This sub-section details the dynamic sequence matching approach of ConvSequential-SLAM. As discussed in sub-section 3.2.6, our technique creates a dynamic list of query images, i.e., the length of the query sequence list will vary for different sets of query images. During the matching phase, we create a sequential one dimensional (1D) reference list of the same length as the sequential query list. These sequential 1D reference lists are created for all the images in the reference map. Because the size of our reference list is dependent on the sequential query list's length, this simplifies the matching of the query and reference image sequences. The algorithm that retrieves a correct match for a sequence of query images given a reference map can be found in Algorithm 2. The function *Sequence_Matching_Func* in Algorithm 2 takes **K** corresponding pairs (1-to-1 matching) from the query image sequence (Q_Seq) and refer-

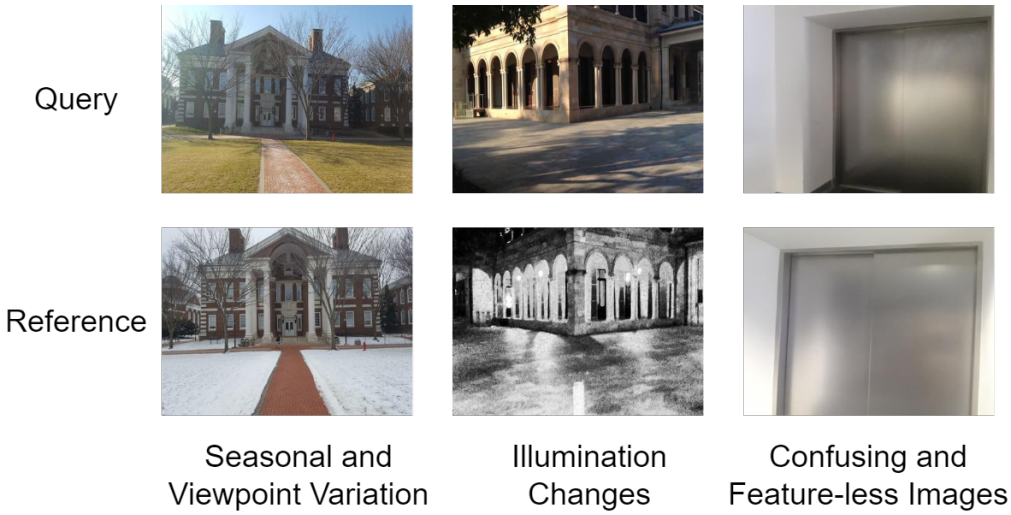


Figure 3.3: The most common changes in the environment are presented here.

ence image sequence (R_Seq) and matches them using Regional Convolutional Matching, as explained in sub-section 3.2.4. The matching score of the query and reference sequences is the arithmetic mean of the matching scores of the pairs within these sequences. This function returns the matching score of the query image sequence and the reference image sequence. Given all the reference images and their corresponding sequences from the reference map, the sequence with the highest matching score is selected as the best match.

3.3 Experimental Setup

3.3.1 Sequential Datasets

To evaluate the proposed technique, we have used four VPR datasets which cover some of the most common viewing conditions in real-world applications, as presented in sub-section 2.7.1 of chapter 2. These are as follows: Gardens Point [39] day-to-day and day-to-night, Nordland [78] and Campus Loop [81]. Apart from using these datasets to show the performance of our technique, we also use the Alderley (night-to-day) dataset solely to show the variation in sequence length due to sequential entropy. Fig. 3.3 shows some sample images representing the most common challenges in VPR.

3.3.2 Utilised VPR Techniques

We compare the performance of ConvSequential-SLAM with other VPR techniques, such as CoHOG [67], HOG [65], CALC [81], HybridNet [13], AMOSNet [13], SeqSLAM [72], RegionVLAD [83], NetVLAD [74] and DeepSeqSLAM [98] on the datasets mentioned in sub-section 3.3.1. We have used SeqSLAM with a sequence length of 5 and 10 images respectively, while DeepSeqSLAM was tested with a sequence length of 10 images only. The remaining VPR techniques are single-image-based and are provided for completeness.

All VPR techniques presented in this study are written in Python, except SeqSLAM[2] which is written in Matlab. The implementations of CALC, HOG, HybridNet, AMOSNet, RegionVLAD and NetVLAD have been used as presented in [7], with source code available within a shared GitHub[3] repository. Both DeepSeqSLAM[4] and CoHOG[5] can be found as GitHub repositories at their respective web addresses. ConvSequential-SLAM was written in Python 2.7 and requires the following Python libraries to run: $cv2$[6], $numpy$[7] and $skimage$[8].

3.3.3 Parameters

In this work, we have used $W1 = H1 = 512$, $W2 = H2 = 16$, $L = 8$ bins, G (can take different values for different query images depending on the scene that is represented), $ET = 0.5$, $IT = 0.9$, $min\_K = 1$, $max\_K\_IG = 15$ and $max\_K = 25$ for ConvSequential-SLAM. These values represent the backbone of our system, as they are responsible for determining the optimal sequence length. The above values were specifically chosen as they provide overall good results in terms of Accuracy, AUC and PCU while also providing comparable results to our static sequence length ($\mathbf{K} = 10$ images) version of ConvSequential-SLAM. An ablation study showing the performance of our technique in terms of accuracy and AUC with various sequence lengths ($1 \leq \mathbf{K} \leq 20$) is provided later in this work.

[2]https://github.com/OpenSLAM-org/openslam_openseqslam
[3]https://github.com/MubarizZaffar/VPR-Bench
[4]https://github.com/mchancan/deepseqslam
[5]https://github.com/MubarizZaffar/CoHOG_Results_RAL2019
[6]https://pypi.org/project/opencv-python
[7]https://pypi.org/project/numpy
[8]https://pypi.org/project/scikit-image

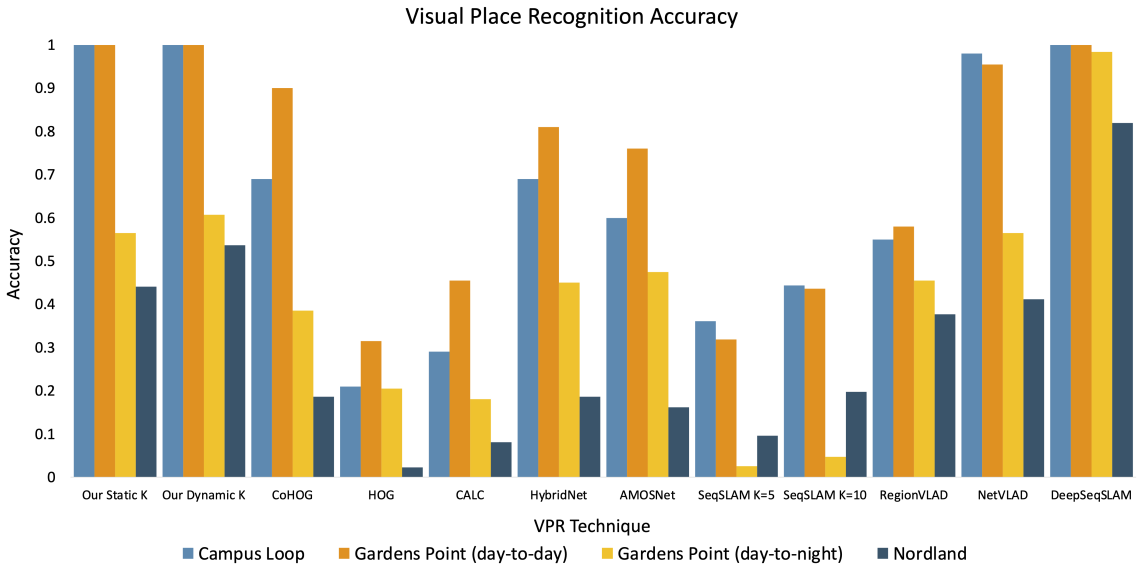


Figure 3.4: The accuracy of ConvSequential-SLAM is compared against the accuracy of other well-established VPR techniques on widely used public VPR datasets.

3.3.4 Performance Metrics

In this chapter, we report the performance of ConvSequential-SLAM utilising several performance metrics, described in sub-section 2.7.2. The first performance metric employed in this work is the accuracy, which represents the percentage of correctly matched images (refer to equation (2.5)). Another important performance indicator employed in this chapter is the AUC, computed by plotting the Precision and Recall at different confidence thresholds utilising equations (2.1) and (2.2). PCU is a relative evaluation metric that combines Precision at 100% Recall (P_{R100}) with the feature encoding time t_e of a VPR descriptor. Thus, this performance metric is employed in this chapter utilising equation (2.4).

3.4 Results and Analysis

In this section, we discuss results from a place matching performance point, in terms of accuracy, AUC and PCU. We also present the performance of ConvSequential-SLAM for various sequence lengths and show how the sequence length varies between one dataset and another. Finally, we show some samples of correctly and incorrectly retrieved query and reference images by our technique for a qualitative insight. For all experiments presented below, we have used a PC equipped with an Intel Core i7-4790k CPU.

Table 3.1: The AUC of VPR techniques on the four datasets.

VPR Technique	**AUC**			
	Campus Loop Dataset	*Gardens Point (day-to-day) Dataset*	*Gardens Point (day-to-night) Dataset*	*Nordland Dataset*
Ours Static K	0.999	1	0.754	0.533
Ours Dynamic K	1	1	0.8	0.6
CoHOG	0.776	0.928	0.43	0.151
HOG	0.301	0.431	0.294	0.036
CALC	0.597	0.738	0.403	0.104
HybridNet	0.889	0.933	0.595	0.214
AMOSNet	0.872	0.907	0.571	0.132
SeqSLAM K = 5	0.273	0.296	0.037	0.059
SeqSLAM K = 10	0.371	0.333	0.071	0.16
RegionVLAD	0.412	0.739	0.642	0.563
NetVLAD	0.998	0.959	0.698	0.733
DeepSeqSLAM	0.999	1	0.952	0.736

3.4.1 Accuracy

This sub-section presents the accuracy results of ConvSequential-SLAM against the performance of other VPR techniques. Fig. 3.4 shows the computed values of accuracy for all techniques, on all 4 datasets. We report in Fig. 3.4 the accuracy of ConvSequential-SLAM using a fixed sequence length of 10 images as well as a dynamic length determined by the system itself. As all the datasets tested contain consecutive images, there is a high possibility that each image is similar to the ones located in its immediate proximity. Therefore, if for any query image, the reference image found to be the best match is in the range ± 2, we consider it as a correct match, except for the Nordland dataset where we use the ± 1 range.

ConvSequential-SLAM achieves state-of-the-art accuracy on Campus Loop and Gardens Point (day-to-day) datasets. Our approach also achieves high place matching performance on the highly conditionally-variant Gardens Point (day-to-night) and Nordland datasets, followed by deep learning-based techniques like NetVLAD, HybridNet and AMOSNet. However, DeepSeqSLAM with a sequence length of 10 images outperforms every other VPR technique on both Gardens Point (day-to-night) and Nordland datasets.

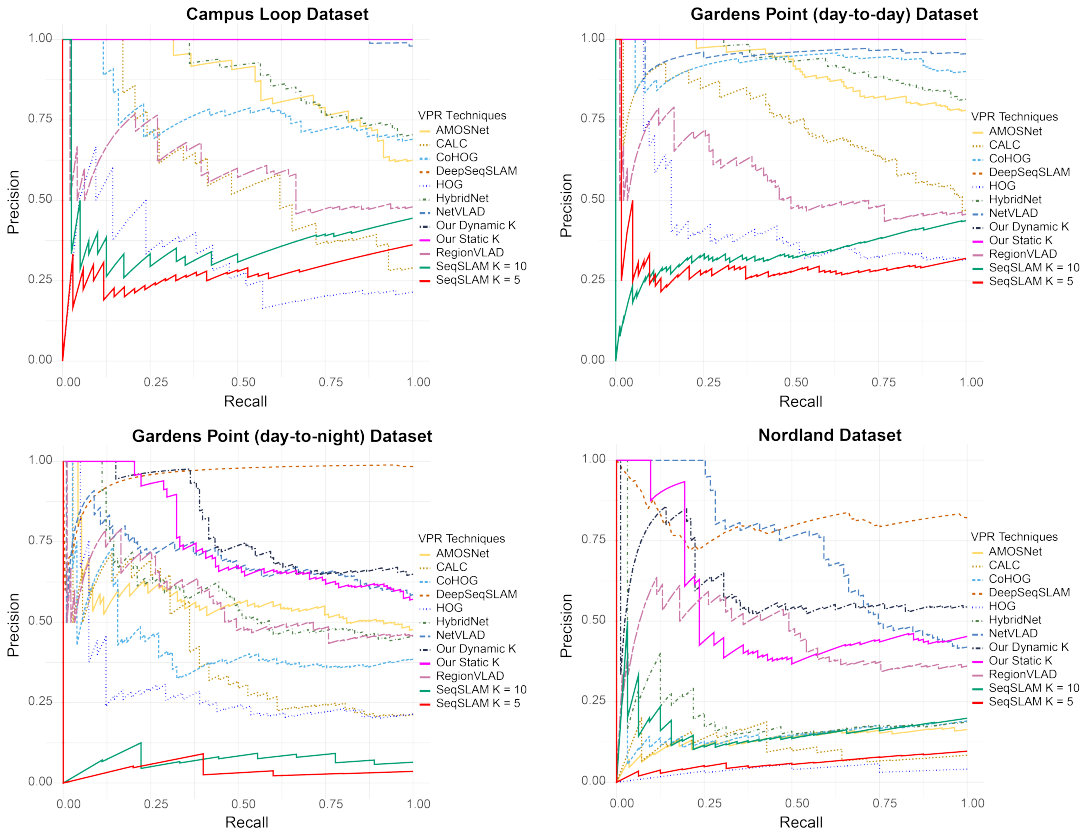


Figure 3.5: The Precision-Recall Curves for all VPR techniques on each of the 4 datasets used in this work are enclosed here.

3.4.2 Area-Under-the-Precision-Recall-Curve (AUC)

The performance of ConvSequential-SLAM in terms of AUC on all datasets is reported in Table 3.1. It achieves good AUC performance on the Campus Loop, Gardens Point (day-to-day), and Gardens Point (day-to-night) datasets. When compared to NetVLAD, our technique achieves better performance on all datasets tested, except on Nordland dataset, as reported in Table 3.1. We can see a small boost in performance between our algorithm using a fixed sequence length of 10 images and a dynamic sequence length respectively. When compared to DeepSeqSLAM, the proposed technique achieves comparable results on both Campus Loop and Gardens Point (day-to-day) datasets. However, DeepSeqSLAM outperforms all VPR techniques on the remaining two datasets (Gardens Point (day-to-night) and Nordland). In Fig. 3.5, we present the Precision-Recall curves of all the VPR techniques tested in our work on all four datasets introduced in sub-section 3.3.1.

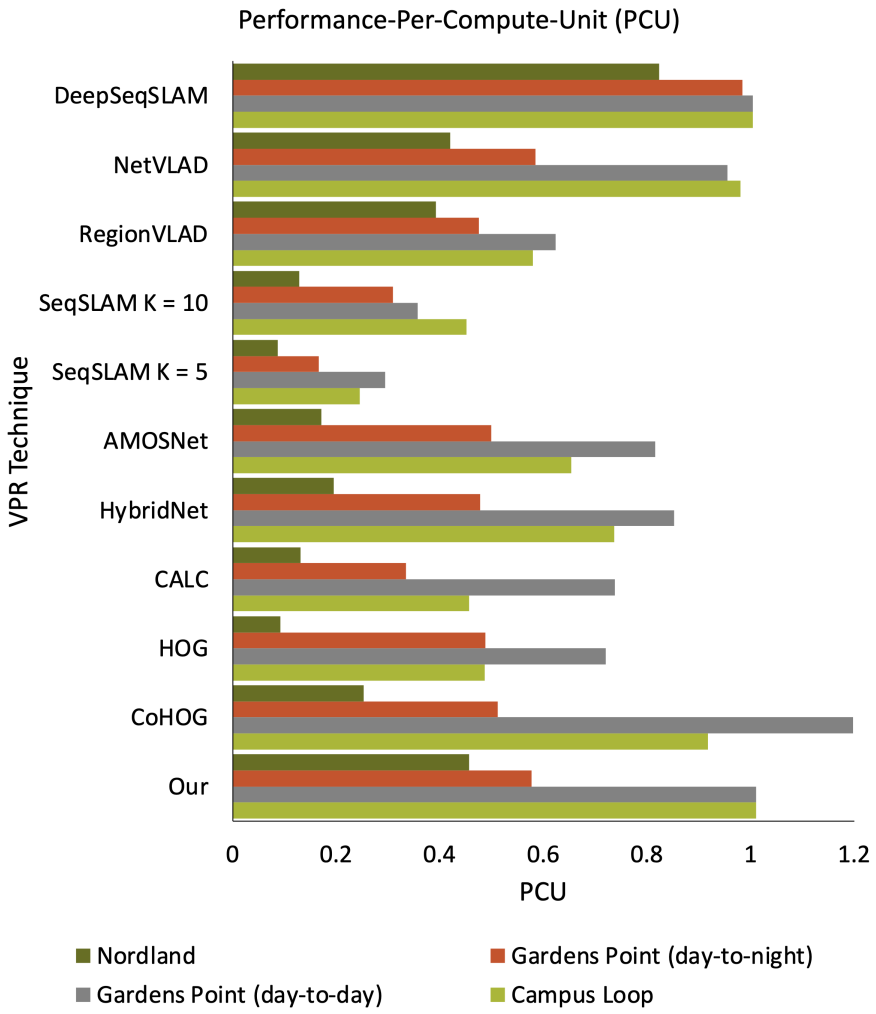


Figure 3.6: The PCU of ConvSequential-SLAM is compared with the PCU of other well-established VPR techniques on all mentioned datasets.

3.4.3 Performance-Per-Compute-Unit (PCU)

Fig. 3.6 presents the PCU of ConvSequential-SLAM using a fixed sequence length of 10 images. Because we match sequences of images instead of single frames, the feature encoding time will also be increased with each image that is part of that sequence, as shown in Fig. 3.7. In this figure, $Ours\ K = 5$ and $K = 10$ represent the feature encoding time of ConvSequential-SLAM using fixed sequences of 5 and 10 images respectively. The feature encoding time for dynamic **K** will vary between the lowest value (that is for a minimum sequence length determined by the information-gain) and the maximum value for a sequence length of 25 images.

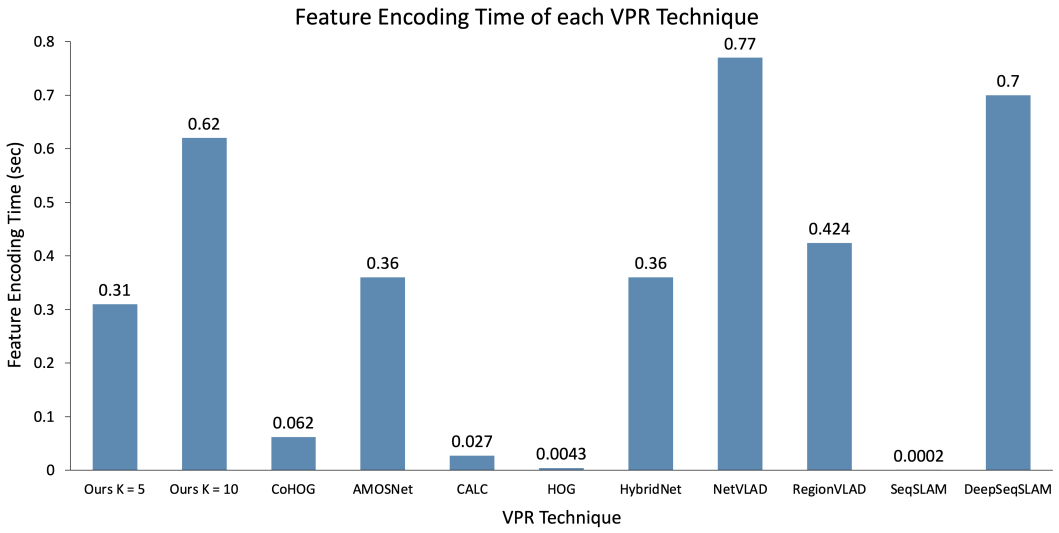


Figure 3.7: The feature encoding times of various VPR techniques are presented in this graph.

In this sub-section, we use the average sequence length computed by our methodology within the dataset for encoding time computation. Even though using a higher sequence length will result in higher encoding times, the performance boost in place matching that is gained greatly benefits ConvSequential-SLAM. This can be seen in Fig. 3.6, where we present the PCU value of our technique against other VPR techniques, on all four datasets. A system that achieves high precision will have a high PCU value, as previously mentioned in sub-section 2.7.2. This is also the case for ConvSequential-SLAM, achieving high PCU values due to its high precision, as seen in Fig. 3.6.

3.4.4 Variation in Sequence Length

It is well known the fact that by incorporating sequence-based filtering into a VPR system, the overall performance is greatly improved. However, sequence matching requires a static sequence length to be provided for each environment that the robotic platform is operating in. Furthermore, this sequence length cannot always be constant as different VPR techniques have different place matching performances. This is an important factor especially for resource constrained platforms, in which the computational intensity of a VPR technique needs to be carefully considered. In this sub-section, we show that ConvSequential-SLAM is successfully able to adapt its sequence length depending on the environment.

For each dataset employed in this study, Fig. 3.8 shows the number of image sequences of length **K** obtained by ConvSequential-SLAM. More specifically, it shows how the sequence

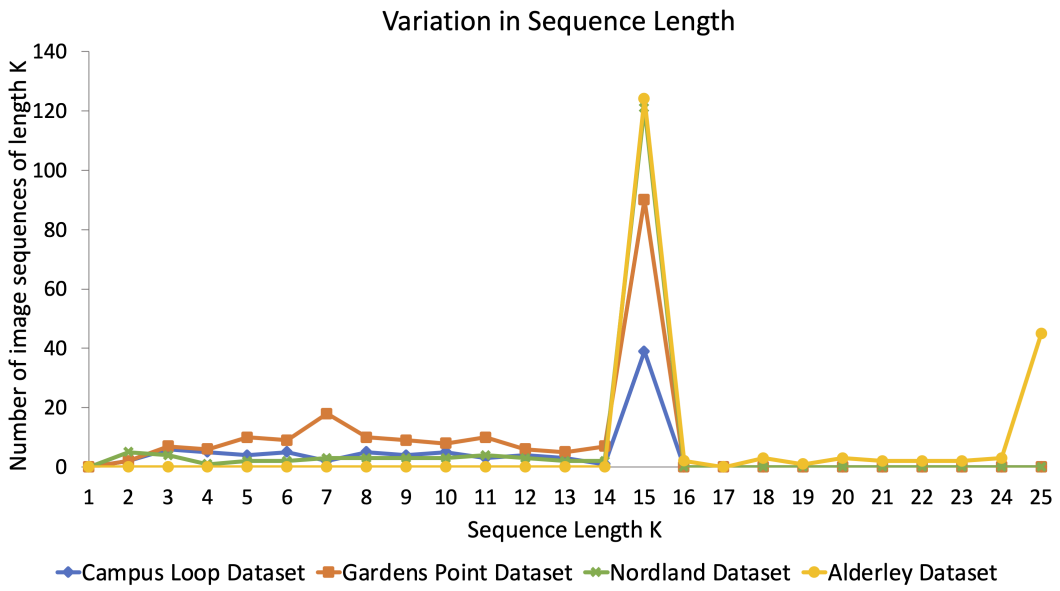


Figure 3.8: The variation in sequence length of ConvSequential-SLAM on all four datasets is shown here.

length changes throughout the entire dataset, and details the exact number of sequences of length **K** created, where $1 \leq \mathbf{K} \leq 25$. It is important to note that by varying both the Entropy Threshold (ET) and Information Threshold (IT) we can achieve lower or higher sequence lengths. Also, it is worth noting that we use *day left* as query images for both Gardens Point (day-to-day) and Gardens Point (day-to-night) datasets, so we only include one instance of the dataset in Fig. 3.8 in order to avoid redundancy.

In addition to the datasets mentioned above, we also use the Alderley (night-to-day) dataset to show how the sequence length is modifying because of entropy. However, all VPR techniques poorly perform on this dataset, because the query images (night images) provide little to no information about the environment. This is due to the poor lighting condition in the environment as well as the presence of rain, which increase the difficulty in place matching.

Because in the Campus Loop, Gardens Point and Nordland datasets the entropy across each dataset is too high, the sequence length would not have increased in most cases, therefore we would end up with non-optimal sequence lengths. By using information-gain resulted from analysing consecutive query images, we are able to increase the minimum sequence length even though the salient information found in any given query image is above the threshold set (e.g. $ET \geq 0.5$). However, in contrast with the previously mentioned datasets

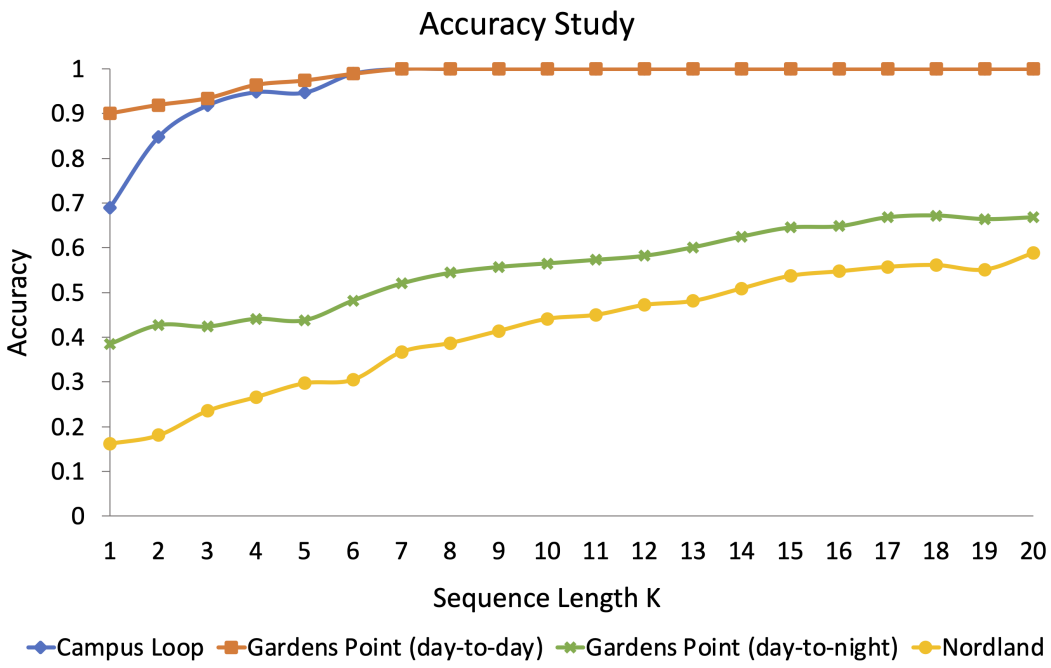


Figure 3.9: The ablation study showing the accuracy of ConvSequential-SLAM when utilising a fixed sequence length ($1 \leq K \leq 20$).

where the sequence length would not increase due to high information content in query images, on the Alderley dataset query images (night images) do not contain salient information due to poor illumination, therefore the sequence length will increase up to the maximum sequence length of 25 as shown in Fig. 3.8.

Using entropy is particularly helpful in scenarios where the query frames do not provide much information (as mentioned above), thus increasing the sequence length allows better chances of finding the correct reference image for any given query image. On the other hand, in cases where the information content of multiple sequences of query images is too high in a dataset (such as Campus Loop, Gardens Point and Nordland datasets), non-optimal sequence lengths will be achieved if entropy alone is used. Therefore, information gain is used to establish the lower bounds of the sequence length needed to achieve optimal results.

3.4.5 Ablation Study

Fig. 3.9 and Fig. 3.10 present the performance of our approach in terms of accuracy and AUC values, when using a fixed sequence length **K** between 1 and 20 images respectively. Increasing the value of **K** leads to an increase in both accuracy and AUC performance. In both

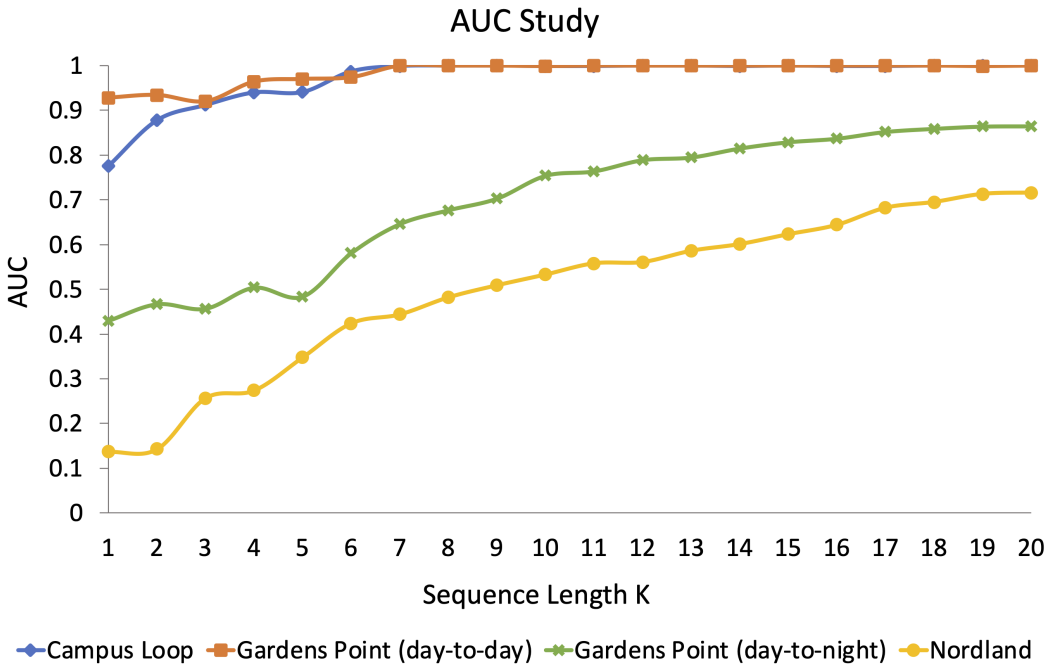


Figure 3.10: The ablation study showing the AUC of ConvSequential-SLAM when utilising a fixed sequence length ($1 \leq K \leq 20$).

Campus Loop and Gardens Point (day-to-day) datasets, a sequence length of $\mathbf{K} = 7$ images will result in the best performance, whilst a higher sequence length is needed to achieve desirable results for the remaining two datasets: Gardens Point (day-to-night) and Nordland.

3.4.6 Exemplar Matches

Fig. 3.11 shows some correctly matched sequences of query and reference images, taken from each dataset. Some failure cases for the proposed technique are shown in Fig. 3.12. These are primarily due to the presence of confusing features coming from trees and vegetation that can be found in most images throughout the Nordland dataset, increasing the difficulty in place matching.

3.5 Summary

As previously discussed in chapter 2, sub-section 2.5, matching sequences of images instead of single camera frames can enable better place matching performance, especially in challenging conditions such as severe illumination and seasonal changes. However, sequence-based

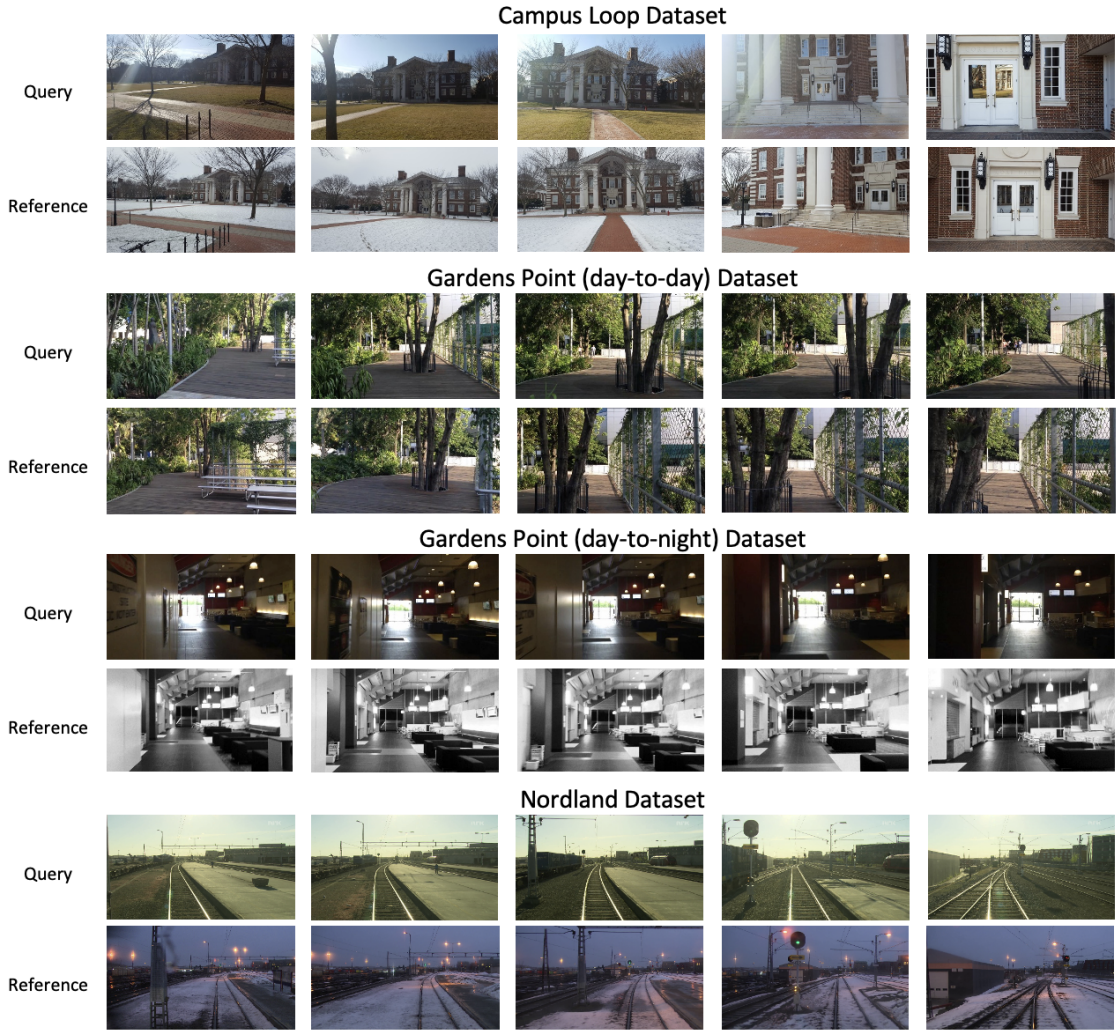


Figure 3.11: Some correctly matched sequences of query and reference frames.

filtering adds more computational complexity to the VPR process, which paired with a deep-learning-based VPR technique can severely restrict its applicability on resource constrained platforms. Moreover, as each VPR technique achieves distinct levels of place matching performance in different scenarios, a constant sequence length can either result in sub-optimal VPR performance or an unnecessary increase in the computational load of the system.

To overcome the limitations of utilising a constant sequence length and the demanding computational resources required to run a deep-learning-based VPR technique, this chapter presents ConvSequential-SLAM, a dynamic sequence-based and training-less VPR technique for changing environments which achieves comparable place matching performance with well-established deep-learning-based VPR techniques on public VPR datasets that contain

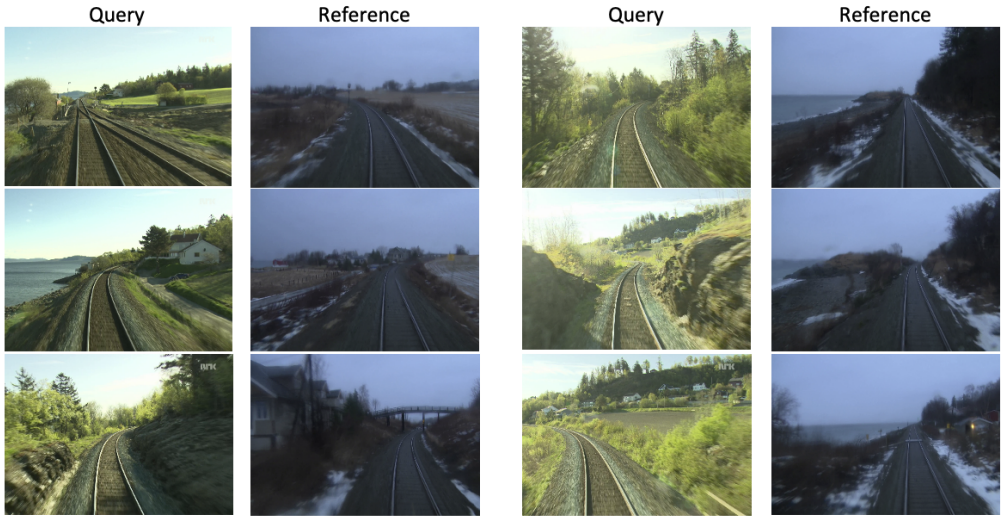


Figure 3.12: Some incorrectly matched query and reference frames.

both viewpoint and appearance variations. We developed a new analysis based on information-gain and entropy to formulate a dynamic sequence length for optimal VPR performance in changing environments. Moreover, the sequence-matching approach proposed in this chapter is agnostic to the underlying VPR technique, hence it can be incorporated in both handcrafted and deep-learning-based VPR techniques. Thus, the following chapter proposes an in-depth study on the effects of sequence-based filtering on top of single-frame-based VPR techniques.

Chapter 4

Sequence-Based Filtering for Visual Route-Based Navigation

An emerging trend in VPR is the use of sequence-based filtering methods on top of single-frame-based place matching techniques for route-based navigation. The combination leads to varying levels of potential place matching performance boosts at increased computational costs. This raises a number of interesting research questions: How does performance boost (due to sequential filtering) vary along the entire spectrum of single-frame-based matching methods? How does the sequence matching length affect the performance curve? Which specific combinations provide a good trade-off between performance and computation? However, there is a lack of previous work investigating these fundamental questions, whilst most of the sequence-based filtering work to date has been used without a systematic approach. To bridge this research gap, this chapter conducts an in-depth investigation of the relationship between the performance of single-frame-based place matching techniques and the use of sequence-based filtering on top of these methods. The sequence-based filtering schema presented in chapter 3 is employed to perform the above mentioned experiments, as it is agnostic to the underlying single-frame technique. This chapter also analyses individual trade-offs, properties and limitations for different combinations of single-frame-based and sequential techniques. The experiments conducted in this study demonstrate the benefits of sequence-based filtering over the single-frame-based approach using various VPR techniques.

4.1 Introduction

In recent years, it has been shown that sequence-based VPR systems such as [72, 98, 133, 141] and [34] can achieve good performance in changing environments. Thus, an almost parallel track has emerged where sequence-based techniques have been shown to outperform single-frame-based techniques. More importantly, the benefits presented by sequential information are generally extendable to most non-learning and learning-based VPR techniques albeit at varying levels and costs. Therefore, it is critical to understand the properties of sequential-based filtering, its trade-offs and how to deploy them on single-frame-based VPR techniques for designing better VPR systems.

To the best of our knowledge, there is no previous work that has examined this important problem in a systematic way (such as performance boost variations due to sequential filtering along the entire spectrum of single-frame-based VPR methods, the effects of sequence length on performance, performance-computation trade-off etc.). To bridge this research gap, this chapter investigates the relationship between the performance of single-frame-based, learnt and non-learnt VPR methods, and the use of sequence-based filtering on top of these methods. In particular, this chapter introduces sequential information into a number of VPR techniques to improve conditional invariance and shows that sequence matching takes a poorly performing single-frame-based VPR technique and improves its performance. While sequence matching has a positive effect on VPR accuracy, it increases the time required to perform VPR. This chapter examines the effects of different sequence lengths on the resulting performance boost and determines the optimal combinations between different VPR techniques and sequence lengths, taking into consideration both the performance and computational load of each system. We found that high-precision VPR systems slightly improve their performance from introducing sequential-based filtering. On the contrary, less accurate but lightweight techniques can receive a significant boost in their VPR accuracy, whilst in some cases also keeping the matching time shorter than state-or-the-art techniques. For example, CALC outperforms NetVLAD on Campus Loop dataset using a sequence of 16 images while taking about 78% of the time to perform VPR. The sequence-based filtering schema employed is summarised in Fig. 4.1 and discussed in-depth in sub-section 4.2.2.

In summary, our work provides the following contributions:

- The application of sequence-based filtering on top of single-frame-based methods is investigated. In particular, we analysed the VPR performance improvement and the computational effort required to execute VPR using a sequence compared with the single-

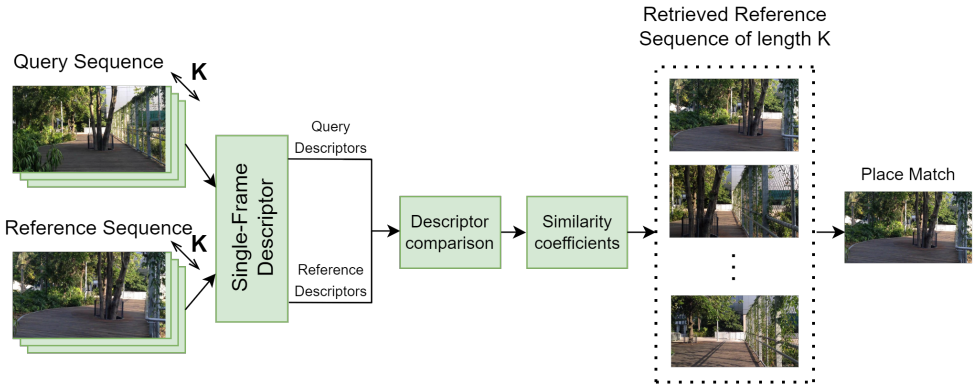


Figure 4.1: The sequence-based filtering schema employed is presented.

frame approach.

- The trade-off between VPR accuracy and computational efficiency is examined, showing how lightweight techniques can replace state-of-the-art descriptors to perform VPR more efficiently, without any loss in accuracy.

4.2 Methodology

This section presents the approach taken for evaluating the boost in performance resulted from introducing sequence-based filtering on top of single-frame-based techniques. To enable the comparison of different VPR descriptors, the sequence filtering schema presented in chapter 3 has been employed, as it is agnostic to the underlying single-frame technique. This approach combines the outcome of single-frame matching operations into a scalar representing the similarity between sequences of images representing the places to match. The sub-sections below provide details on sequence-based filtering and the evaluation criteria used to assess the impact of sequence-based filtering on single-frame-based VPR techniques.

4.2.1 Single-Based Image Matching

For any given query image (e.g. a frame taken from a robot's camera), the main goal of a VPR technique is to retrieve the most representative reference image (the matching place) from the database. This is done by comparing each query image with all the stored database images in such a way that each time a query and reference image are matched together, a similarity score is computed. For any given query image, the reference image with the highest score is chosen as the best match.

Algorithm 3: Query and Reference Descriptor Comparison.

Given: Query Descriptor (Q_F)

Given: Map of Reference Descriptors (R_M)

INITIALISE (array of 0s): score_array[length(R_M)]

iterator = 0

for R_F ***in*** R_M **do**

    score = Cosine_Similarity(Q_F, R_F)

    score_array[iterator] = score

    iterator = iterator + 1

Best_Match = Max(score_array)

The feature descriptor computed by a VPR technique for a query image *Q* is denoted as Q_F, for a reference image *R* as R_F whilst the list containing the reference descriptors for the entire map as R_M. The similarity between two image descriptors (Q_F and R_F) is determined using the cosine [79]:

$$s = \frac{Q_F R_F}{||Q_F||\ ||R_F||} \tag{4.1}$$

The single frame-matching schema requires that Q_F is compared with every R_F from R_M. Thus, for any *N* images in a dataset, a set of similarity scores *S* is created as follows:

$$S = \{s_1, s_2, s_3, ..., s_N\} \tag{4.2}$$

Where $s \in \mathbb{R}$ and *s* in range [0,1]. Higher the score, higher the similarity between two image descriptors.

For each query image *Q*, a new set of similarity scores *S* is created containing the values for that particular frame. Once the similarity coefficients have been computed, the reference image with the highest value ($s \in S$) is regarded as the matching place for Q_F.

Algorithm 3 presents the entire matching process for a query image descriptor Q_F and the map, R_M. The matching score (calculated as in equation (4.1)) of each query-reference pair is stored in a 1D array entitled *score_array*. Once a similarity score has been generated for every R_F (from R_M), the maximum value from the *score_array* is retrieved, and thus, the most representative reference image for Q_F is selected as the best match.

4.2.2 Sequential-Based Filtering

In contrast to the single-image matching process previously mentioned, sequential-based filtering allows a VPR technique to match sequences of query and reference frames. Fig. 4.1

summarises the sequence-based filtering schema employed. First, sequences of query and reference images of constant length are created. A similarity value is generated each time a sequence of query and reference images are matched together. Thus, for each sequence of query images, the reference sequence with the highest similarity score is selected as the most representative match. The most important steps for introducing sequential-based filtering on top of single-frame-based VPR techniques are presented below:

Creating the Image Sequence

For any given query image q_i, the sequence of **K** consecutive images is built as follows:

$$q_i \quad q_{i+1} \quad q_{i+2} \quad \cdots \quad q_K \tag{4.3}$$

Where q_i is the query image for which the sequence is built, q_K is the last query image that is part of the given sequence, and **K** is the total number of images that forms each sequence.

Similarly to (4.3), the reference images are organised in sequences (formed with an offset of 1 image) as presented in equation (4.4):

$$\begin{matrix} r_1 & r_2 & r_3 & \cdots & r_K \\ r_2 & r_3 & r_4 & \cdots & r_{K+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{N-K+1} & r_{N-K+2} & r_{N-K+3} & \cdots & r_N \end{matrix} \tag{4.4}$$

The application of equation (4.4) results in *N - **K** + 1* image sequences, where *N* is the total number of images in the dataset and **K** is the sequence length. Using higher sequence lengths will lead to less images to be searched for, as no new image sequences of length **K** can be created when we approach the end of the dataset. For this reason, the number of sequences created depends solely on the value of the selected sequence length ($2 \leq \mathbf{K} \leq N$) as shown below:

$$No.\ of\ Seq\ Created = N - \mathbf{K} + 1 \tag{4.5}$$

Once the query and reference sequences are created, the sequence matching is performed.

Sequence Matching

All query and reference features are initially computed and stored in two separate 1D lists: Q_F and R_F. *perform_VPR* in Algorithm 4 has two main functions, more specifically creating

Algorithm 4: Creating Query and Reference Sequences.

Given: Total Number of Query Images

Given: Total Number of Reference Image

K = image sequence length

for *i* ***in range*** *(total_Query_Images -* ***K*** *+ 1)* **do**
    ref_matching_scores = []
    for *j* ***in range*** *(total_Ref_Images -* ***K*** *+ 1)* **do**
        score = *perform_VPR(Q_F, R_F, i, j)*
        ADD score to ref_matching_scores
    Best Match = Max (ref_matching_scores)

the query and reference image sequences of constant length **K** from Q_F and R_F (presented in sub-section 4.2.2) and image sequence matching.

The *perform_VPR* function firstly takes the indices (*i* for query images and *j* for reference images) from Algorithm 4 in order to determine for which query and reference image the sequences will be created. Starting from the *i-th* image, the *perform_VPR* function creates sequences by adding consecutive images until the required sequence length **K** has been obtained. The same process is repeated for every reference image, starting with the *j-th* image. This process is presented in Algorithm 5, which represents the *perform_VPR* function.

For every given query image sequence previously created, *perform_VPR* searches for the most representative reference image sequence. Algorithm 5 presents the process of matching a sequence of query and reference images, generating **K** similarity values (*score*) for each query-reference pair that are part of the matched sequences. The similarity or matching score of any query-reference image sequences (*sequential_score*) is calculated as the arithmetic mean of the matching scores of the pairs within these sequences. Thus, the matching score for a sequence of images of length **K** is computed as:

$$s' = \frac{\sum_{i=1}^{K} s_i}{K} \tag{4.6}$$

Where s_i represents the matching score for each query-reference pair with index *i*. The matching score *s'* has values in range [0,1], with a higher score denoting a better similarity between two sequences of query and reference frames. Thus, for each query image sequence, the reference sequence with the highest score is selected as the most representative match. This can be seen in Algorithm 4, where for any given query image sequence, the matching scores of all reference image sequences are stored in a list, namely *ref_matching_scores*. The

Algorithm 5: The *perform_VPR* function is presented here.

Given: List of Query Descriptors (Q_F)
Given: List of Reference Descriptors (R_F)
Given: Query Image Number (*i_query*)
Given: Reference Image Number (*j_ref*)
K = image sequence length
sequential_score = 0
i = i_query
j = j_ref
while *i* $<$ *i_query* + ***K*** *and j* $<$ *j_ref* + ***K*** **do**
    score = Cosine_Similarity(Q_F[i], R_F[j])
    sequential_score = sequential_score + score
    i = i + 1
    j = j + 1
sequential_score = sequential_score / **K**

maximum score from this list is taken as the best match for that given query image sequence.

When analysing a query image q_i, we take into account the sequential information provided from using consecutive images, thus the next ***K*** - *1* images are also analysed as part of q_i's image sequence. For this reason, the first reference image that is part of the sequence with the highest score is retrieved as being the best match for its corresponding query image.

4.3 Experimental Setup

This section discusses the performance metrics employed, the VPR techniques utilised to generate our results and the sequential datasets used in this work.

4.3.1 Employed Performance Metrics

Similarly to chapter 3, we assess the VPR performance of various VPR techniques (presented in sub-section 4.3.2) utilising several performance indicators described in sub-section 2.7.2. These are as follows: AUC, computed utilising the Precision and Recall (refer to equation (2.1) and (2.2), respectively); accuracy, representing the percentage of correctly matched images, computed utilising equation (2.5); PCU, calculated as in equation (2.4).

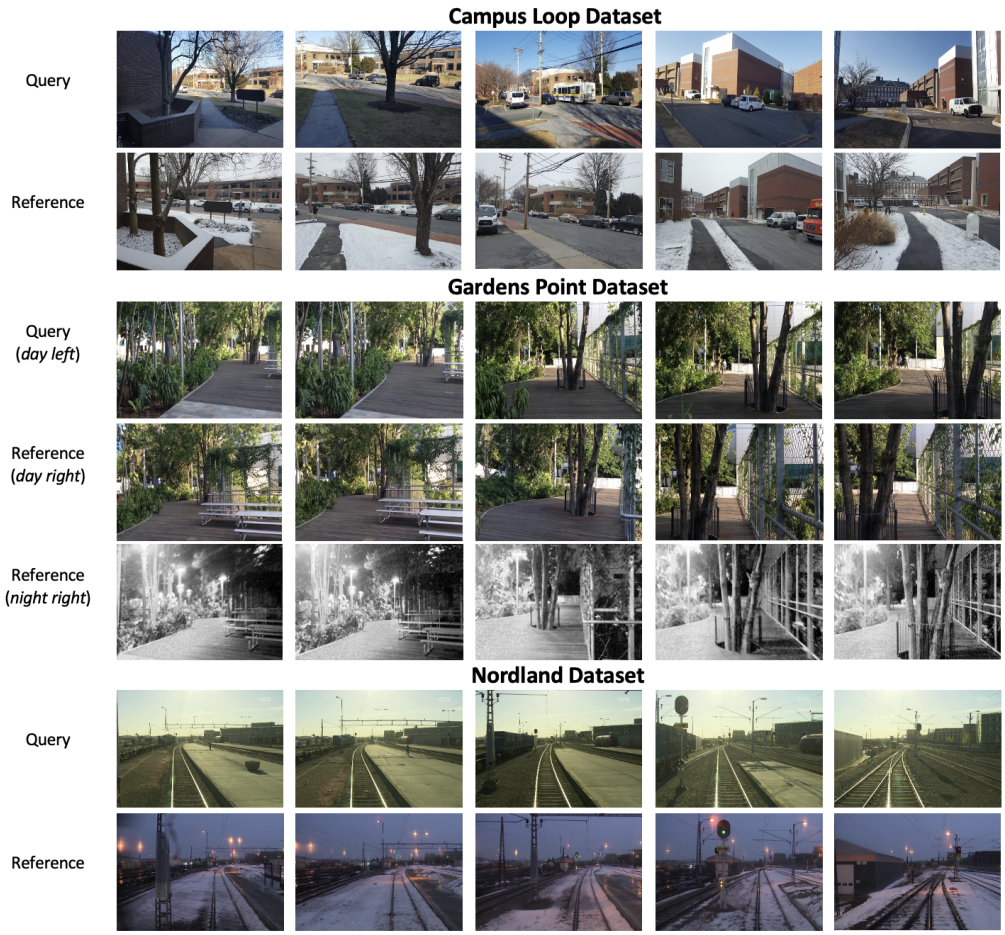


Figure 4.2: Sample sequence of images taken from each of the 4 datasets.

4.3.2 Utilised VPR Techniques

In this work, sequence-based filtering is introduced into a number of state-of-the-art VPR techniques, namely HOG [65], CALC [81], AMOSNet [13], HybridNet [13] and NetVLAD [74]. Single-frame-based implementation of Zaffar *et al.* [7] is used for all 5 aforementioned VPR techniques. In sub-section 4.4, comparative results based on the employed performance metrics for these VPR techniques are presented, along with discussion of the benefits and trade-offs of sequence-based filtering.

4.3.3 Utilised Sequential Datasets

For this study, four sequential VPR datasets are used as presented in sub-section 2.7.1, namely Campus Loop [81], Gardens Point [39] day-to-day and day-to-night and Nordland [78]. Fig. 4.2 shows sample images taken from each dataset.

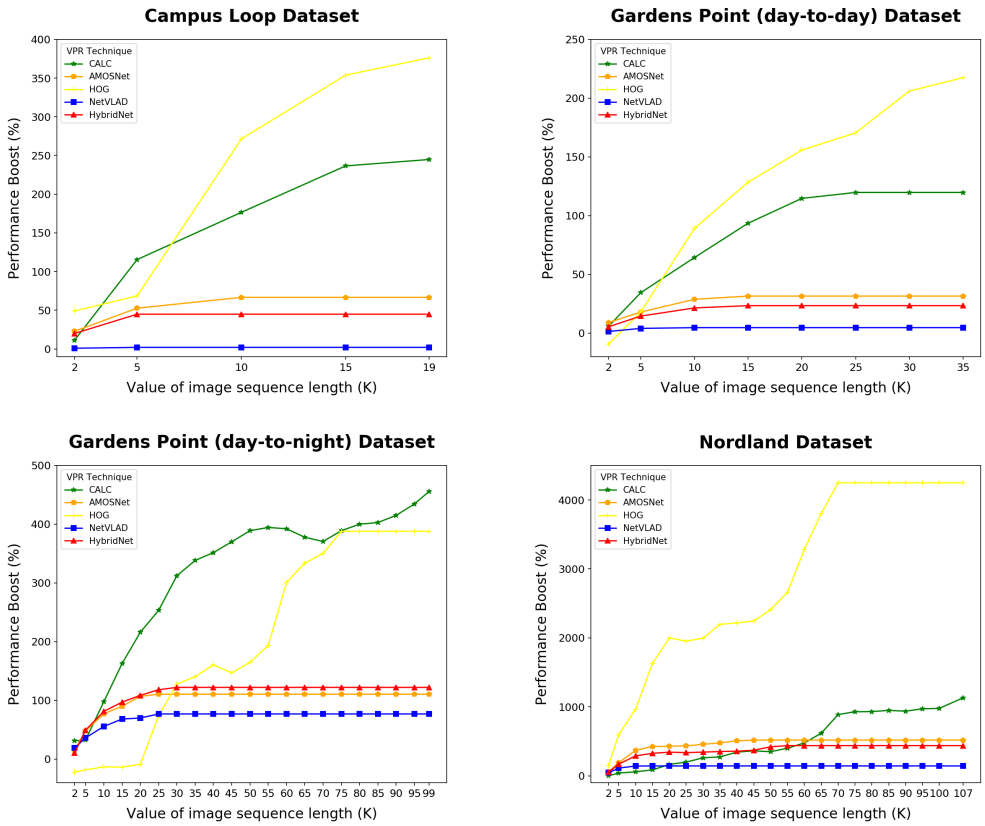


Figure 4.3: The performance boost (%) of sequence matching performance in comparison to the single-frame-matching performance of all VPR techniques on the datasets mentioned in sub-section 4.3.3.

4.4 Results and Analysis

In this section, we focus our attention on understanding how and when to use sequence-based localisation/place matching, its strengths/downsides and the most appropriate performance metrics that can be used in order to test the efficacy of such VPR systems. We present the results for sequence-based filtering when used on top of the VPR techniques mentioned in sub-section 4.3.2. We also present the computational effects of sequential-based filtering and discuss the benefits and trade-offs. For all experiments presented below, we have used a PC equipped with an Intel Core i7-4790k CPU.

4.4.1 Place Matching Performance

In Fig. 4.3 we present the performance boost achieved by different VPR techniques for several sequence lengths. The maximum value of **K** presented in Fig. 4.3 corresponds to the value

Table 4.1: The sequence length K required for each VPR technique to reach maximum place matching performance (100% accuracy) on each of the 4 datasets.

Dataset	VPR Technique				
	NetVLAD	HOG	CALC	AMOSNet	HybridNet
Campus Loop	3	19	16	8	5
Gardens Point (day-to-day)	6	35	23	14	13
Gardens Point (day-to-night)	25	75	99	21	30
Nordland	11	71	107	45	55

required for each method to achieve 100% accuracy. Those **K** values are summarised in Table 4.1 for every VPR technique and dataset. The performance boost in Fig. 4.3 is calculated as the percentage increase between the accuracy of the sequence-based and the single-image version of the same VPR technique. It is evident from Fig. 4.3 that the addition of sequential filtering to a given single-frame-based VPR technique mostly improves the overall place matching performance of that technique. This suggests that by increasing the sequence length of a VPR technique, we will achieve better place matching performance. HOG achieves the highest performance boost on all datasets except Gardens Point day-to-night. VPR techniques such as AMOSNet and HybridNet have a substantial increase in performance using a considerable shorter sequence length (**K**) than simpler VPR techniques, such as CALC or HOG, on Gardens Point day-to-night and Nordland. The reason behind this is that CNN-based VPR techniques such as AMOSNet and HybridNet are designed and trained to deal with drastic changes in the environment, while simpler techniques such as HOG are only able to deal with moderate viewpoint and illumination changes. We further discuss this topic in sub-section 4.4.2. However, VPR techniques which already achieve close-to-ideal matching performance, such as NetVLAD, do not benefit much from using an increased sequence length on certain datasets, such as on the Campus Loop and Gardens Point day-to-day dataset, where the performance boost of the system is negligible. This is mainly because CNNs such as NetVLAD, are successfully able to handle the viewpoint, seasonal and illumination variations that can be found in these datasets, without requiring an increased sequence length. This observation is important as using sequences instead of single images has computational drawbacks and should be avoided where unnecessary. We expand on this further in sub-section 4.4.3 and 4.4.4.

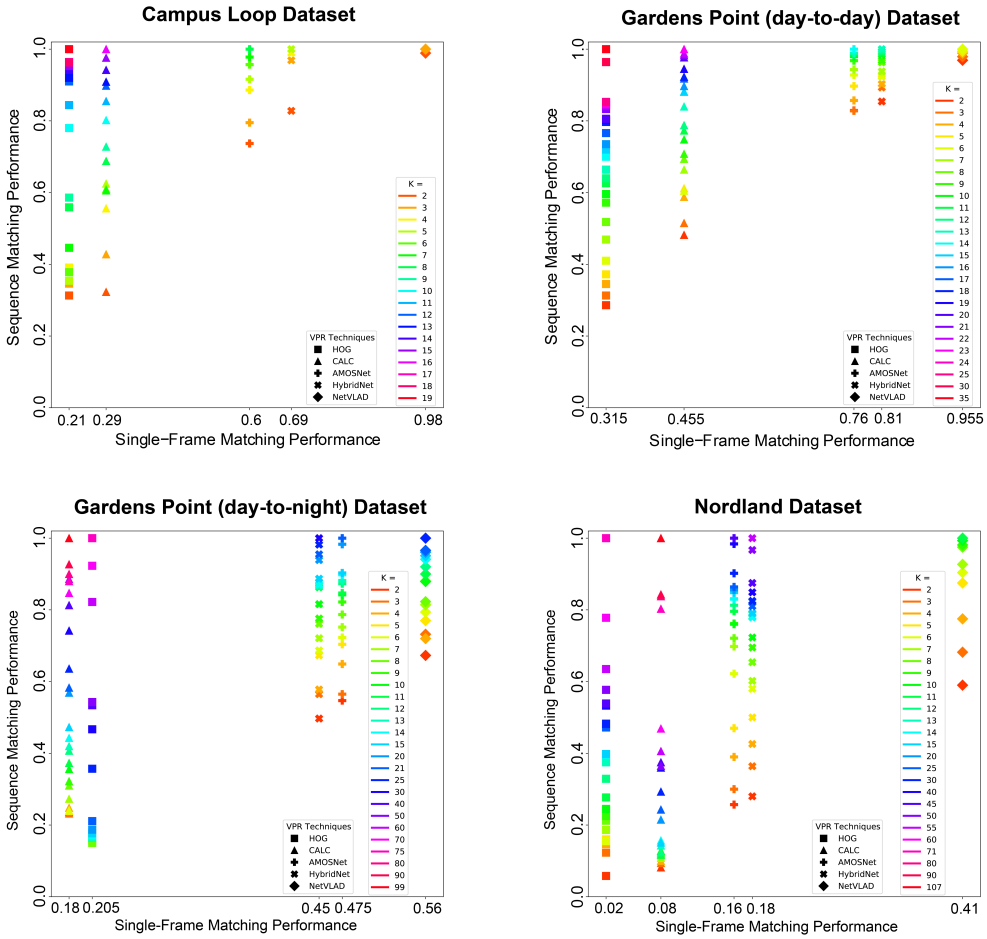


Figure 4.4: The single-frame matching performance compared to the sequence matching performance for all 5 VPR techniques on all 4 datasets.

4.4.2 Performance-Boost Variations

Fig. 4.4 presents the single-frame matching performance (x-axis) of each VPR technique in terms of accuracy and compare it with the sequence matching performance (y-axis) of the same VPR techniques. Plotting the performance variations in this manner helps us to understand the amount of compression and expansion in performance boosts given sequence length variations for different VPR techniques, while also putting it on par with the single-image retrieval performance.

A common observation in existing literature has been that sequential-filtering mostly helps with introducing conditional invariance [72], however, the results obtained on Gardens Point day-to-day shown in Fig. 4.4 demonstrate that it also greatly helps in viewpoint variant, con-

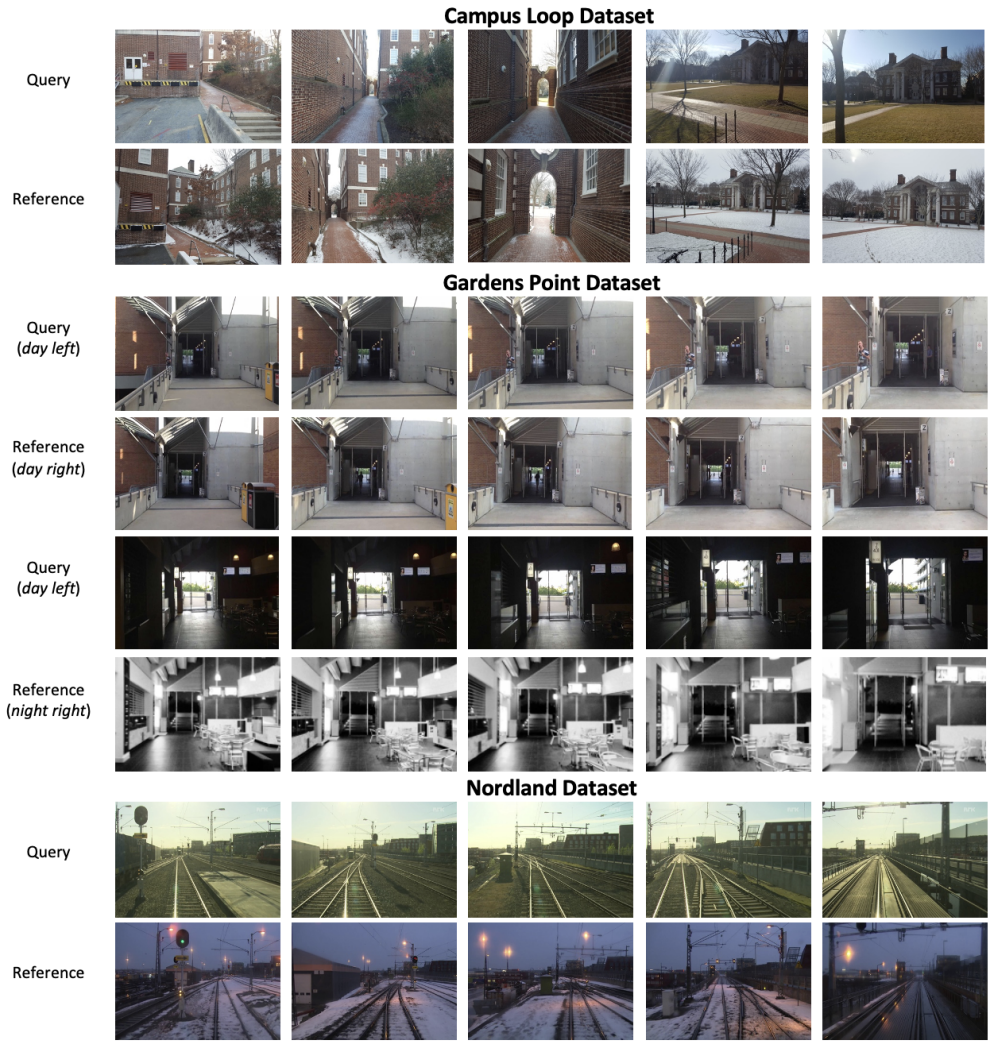


Figure 4.5: Some correctly matched sequences of query and reference images taken from each dataset used.

ditionally invariant scenarios. The performance boost of each VPR technique is directly linked to the severity of the environmental changes (and their effects on the scene appearance) in the dataset. The benefits of sequential-filtering are clearly enjoyed extensively by most techniques on datasets (Campus Loop and Gardens Point day-to-day) with less conditional changes than datasets (Nordland and Gardens Point day-to-night) with extreme conditional changes. Fig. 4.5 shows a sequence of correctly matched query and reference images taken from each of the 4 datasets.

In contrast to the observations made above, the performance improvement of HOG (refer to Fig. 4.3) is inconsistent on the Gardens Point day-to-night dataset (for sequence lengths

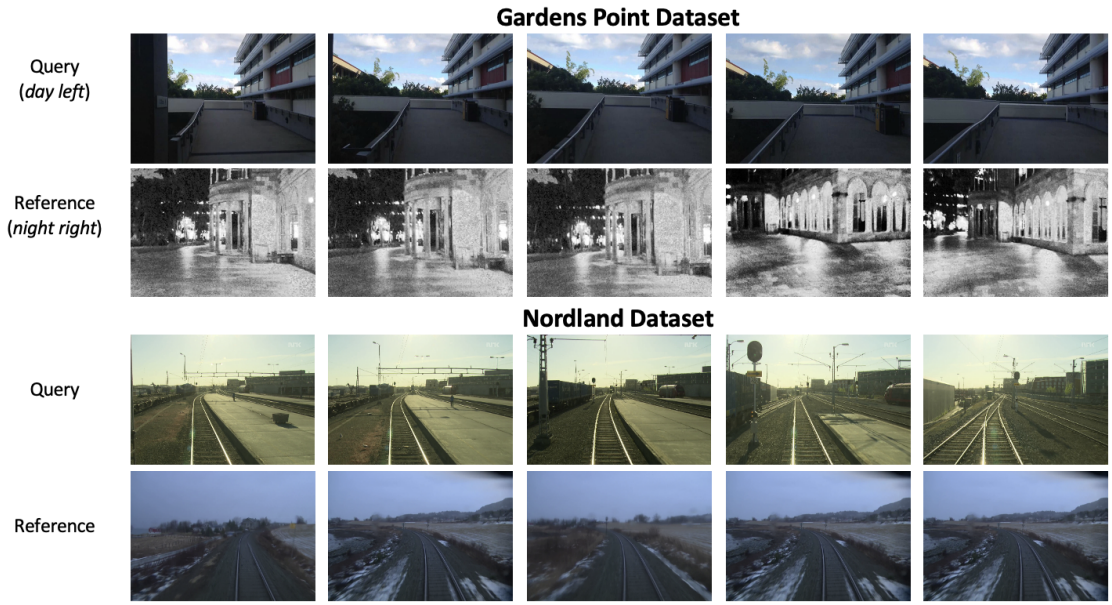


Figure 4.6: Some incorrectly matched sequences of query and reference images taken from Gardens Point day-to-night and Nordland datasets.

of $2 \leq \mathbf{K} \leq 20$), where the single-frame performance of this technique achieves similar or better place matching performance compared to that of the sequence matching performance. The presence of extreme viewpoint variation, illumination variation and also the presence of statically-occluded frames in the Gardens Point day-to-night dataset may affect the performance of this technique. Similarly, the improvement in the performance gained by using sequential-based matching for CALC is more limited (thus requiring a longer sequence length $\mathbf{K}$ to reach maximum accuracy) when compared to other techniques on the Nordland dataset due to the presence of viewpoint and seasonal variation, as seen in Fig. 4.4. On this dataset, even with the addition of sequential-based matching, both HOG and CALC achieve lower results than more complex VPR techniques such as NetVLAD. These results are primarily due to the nature of the dataset, which contains a large number of confusing features, primarily coming from trees and vegetation. On the other hand, the night images from Gardens Point contain a lot of noise (pepper noise) which drastically decrease the place matching performance of light-weight systems such as HOG. We show in Fig. 4.6 some sequences of incorrectly matched query and reference images taken from both Gardens Point day-to-night and Nordland datasets. In such scenarios, evidently it is better for a system to switch to more sophisticated and invariant techniques, such as NetVLAD and HybridNet, even at the expense of higher computational needs.

Table 4.2: Feature encoding times of different VPR techniques.

VPR Technique	Feature Encoding Time (sec)
AMOSNet	0.36
CALC	0.027
HOG	0.0043
HybridNet	0.36
NetVLAD	0.77

In summary, some example cases where using a higher sequence length for a simple, handcrafted VPR technique (such as HOG) is beneficial are laterally viewpoint variant and seasonally variant (but under similar illumination) scenes, e.g. driving a car in a different lane on a previously visited road in a different season. The increasing trend in performance of the HOG technique can be clearly seen in both Fig. 4.3 and Fig. 4.4, for the Campus Loop and Gardens Point day-to-day datasets. However, for platforms that can have 3D or 6-DOF viewpoint changes, e.g. drones, UAVs etc., deep-learning-based techniques should be used instead of simple techniques with high sequence length, which is also the case for highly illumination/conditionally variant scenes such as those found in the Gardens Point (day-to-night) and Nordland datasets. Our data supports the fact that deep-learning-based VPR techniques are better equipped to deal with these variations, and that they should be used in these scenarios instead of more simple VPR systems. Thus, we propose that having this prior knowledge can lead a system based on an ensemble of sequentially-filtered VPR techniques, which are switched accordingly dependent upon the environmental variation cues. This criteria will ensure that the most appropriate VPR technique is selected in each scenario, thus increasing the place matching performance, possibly at much lower computational costs as discussed in sub-section 4.4.3.

4.4.3 Benefits and Trade-Offs of Sequential-Based Filtering

This sub-section presents the benefits and trade-offs of sequential filtering while also answering key questions.

Computational Effects of Sequential-Based Filtering

Due to the fact that we are matching sequences of images instead of the traditional single-frame approach, the feature encoding time for each VPR technique will be increased by **K**

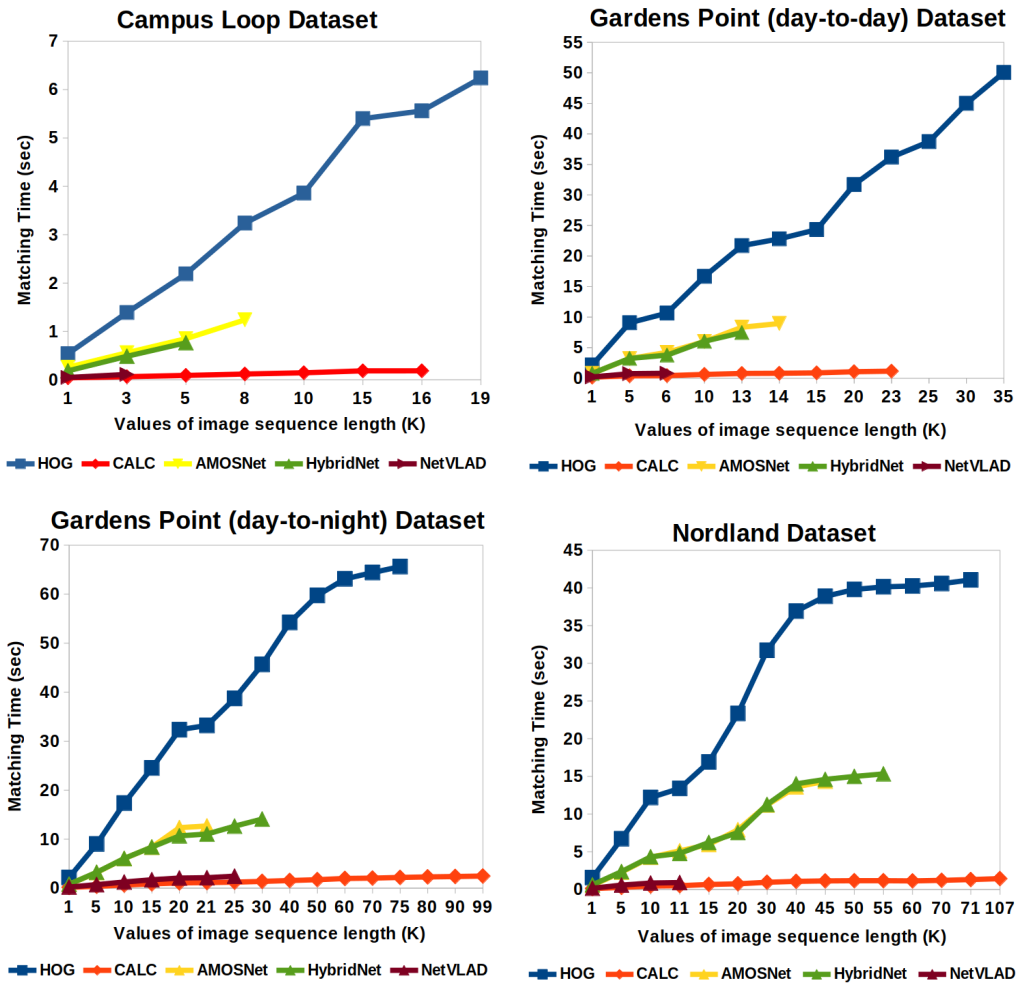


Figure 4.7: The matching time in seconds of each VPR technique on all 4 datasets is presented here. For every VPR technique, we only plot up to the value of the sequence length K that is required to reach 100% accuracy (reported in Table 4.1).

folds. Table 4.2 shows the feature encoding time of the five VPR techniques used in this work without sequential filtering and Fig. 4.7 presents the matching time of each technique. Because neural network-based VPR techniques, such as HybridNet, AMOSNet and NetVLAD already have increased feature encoding times, the addition of sequential filtering will lead to a drastic increase in processing time. Fig. 4.8 shows the PCU of each VPR technique and the computational effects of using multiple sequence lengths. Thus, in both Fig. 4.7 and 4.8, for each VPR technique, we only plot up to the sequence length values (**K**) that are required to achieve 100% accuracy (see Table 4.1). It is important to note that a significant increase in the PCU curves occurs when there is a notable increase in precision compared to the increase

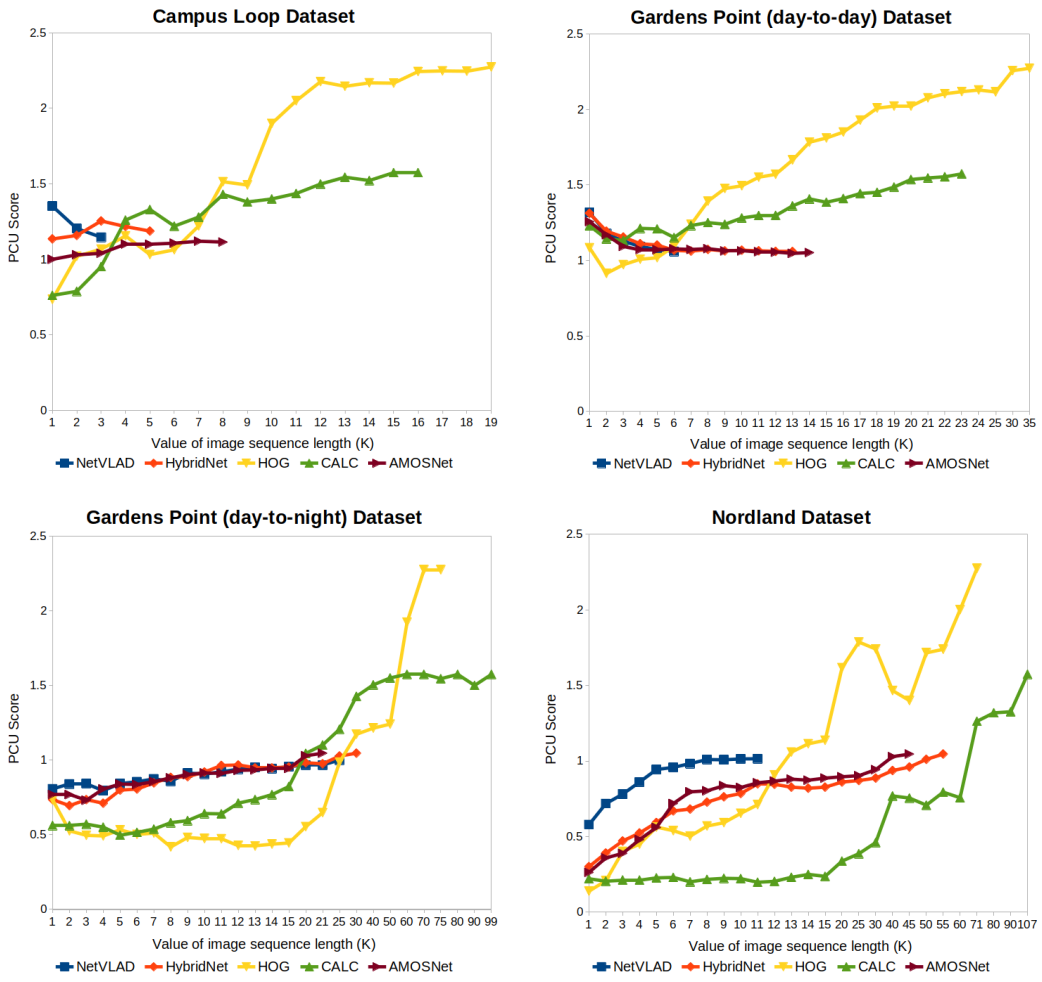


Figure 4.8: The PCU values for each VPR technique on all 4 datasets is reported here. For every VPR technique, we only plot up to the value of the sequence length K that is required to reach 100% accuracy (reported in Table 4.1).

in encoding time. HOG achieves high PCU values due to both its low encoding times and high increase in precision when adding sequential filtering.

Apart from the computational downsides mentioned above, the latency in getting a match as it needs to build up the sequence has to be considered. Furthermore, shifting between two different routes that have not been traversed in that order in the map (switching latency) can increase the computational requirements. Switching latency represents the amount of time required for a VPR system to transition from recognising one location to another [142]. This is crucial for real-time applications that require fast and precise visual place recognition in changing environments. Moreover, the difficulties with variable velocities (solved partially

with more sophisticated search or using odometry information) can lead to further computational constraints. This is especially important for resource constrained platforms as it may restrict its applicability in real world scenarios, due to the high amount of visual information that has to be processed.

Sequence-Based Filtering Vs. Single-Image-Based VPR

The data shows that for a VPR system that has poor performance on a dataset, the addition of sequence-based filtering may greatly improve its performance. Using a longer sequence length will have a higher impact in place matching performance. This is the case for HOG and CALC, which greatly benefit from the addition of sequence-based filtering. On the other hand, the single-image version of NetVLAD already achieves almost perfect results on both Campus Loop and Gardens Point (day-to-day) datasets and thus, the increased computational effects of sequential filtering for just a small gain in place matching performance may not evidently be desirable, as shown in Fig. 4.8. Empirically, the increase in sequence length does not cause any reduction in the place matching performance but mostly yields better performance and therefore, given computational power, it may be desirable to use sequence-based techniques instead of single-image-based techniques.

Performance Benefits Based on Sequential Filtering

As shown in Fig. 4.3 and Fig. 4.4, an increased sequence length for a given VPR technique will lead to higher performance on most datasets tested. However, different VPR techniques will require different sequence lengths (see Table 4.1) depending on the performance of the system on a given dataset. When using sequence-based filtering, the boost in performance can be attributed to several reasons. Primarily, using an increased sequence length increases the chances of finding the best reference image for any given query image which also translates to reduced perceptual aliasing. The increased sequence length also improves the conditional invariance of a VPR technique as shown by our results.

Light-Weight Vs. Deep-Learning-Based VPR Techniques Blended With Sequential-Based Filtering

It is evident that it is indeed possible to use a much simpler, light-weight VPR technique, paired with sequential filtering in order to match or even outperform the effectiveness of

deep-learning-based VPR techniques on certain datasets. We have showed that the performance of a simpler VPR technique, such as HOG, can be drastically increased when using sequence-based filtering with a longer sequence length. The same can be said about CALC, which achieves good results when paired with sequential filtering. Moreover, both VPR techniques have a low feature encoding time, thus greatly benefiting from a PCU standpoint. Using the best VPR techniques (simpler systems with longer sequence lengths or deep-learning-based systems with shorter sequence lengths) for the right dataset will result in an overall better place matching performance, as discussed in sub-section 4.4.2.

4.4.4 Computational Budget

In a real-word scenario where robotic platforms are computationally restrained, it is imperative to achieve the highest VPR performance given computational constraints. In this sense, we show a performance comparison between the best performing single-frame-based VPR technique and the sequence length obtainable by each VPR technique in a given time frame. By adding together the encoding time (t_e) with the matching time (t_m), we obtain the *VPR time* for any technique as follows:

$$t_{VPR} = t_e + t_m \tag{4.7}$$

Using equation (4.7) allows us to make a fair comparison between the performance of each VPR technique and the effects that sequence length has on t_{VPR}. For this reason, t_{VPR} is used as a criterion that helps us determine whether the best performing single-frame-based VPR technique (NetVLAD - refer to Fig. 4.4) can be outperformed by a sequence-matching filtering implementation of other VPR techniques presented in this work. As such, given the t_{VPR} of NetVLAD as computational budget, we present in Table 4.3 and Table 4.4 the maximum sequence length obtainable by each VPR technique in regards to the given time. In case where a VPR technique reaches 100% accuracy before the computational budget is expended, the respective sequence length is reported instead. The values presented in bold represent the VPR technique that has the highest accuracy and the technique that has the lowest t_{VPR}. Apart from the accuracy of a VPR system, we also present the AUC values and the precision at 100% recall (P_{R100}) for that particular sequence length. Moreover, in addition to the VPR techniques presented in sub-section 4.3.2, we also include the results for ConvSequential-SLAM (described in chapter 3). However, as the sequence length of ConvSequential-SLAM is constantly changing, the t_{VPR} for each query image would be different. Hence, we do not include the t_{VPR} in Table 4.3 and Table 4.4, but only present the performance metrics

Table 4.3: Given the t_{VPR} of the best performing single-frame-based VPR technique, we show the maximum sequence length that can be reached by the sequence-based implementation of the remaining VPR techniques, on Campus Loop and Gardens Point (day-to-day) datasets.

	Campus Loop Dataset					
VPR Technique	NetVLAD	HOG	CALC	AMOSNet	HybridNet	ConvSequential-SLAM
K	1	1	16	1	1	dynamic
t_e (sec)	0.77	0.0043	0.432	0.36	0.36	-
t_m (sec)	0.049	0.544	0.21	0.186	0.19	-
t_{VPR} (sec)	0.819	**0.544**	0.642	0.546	0.55	-
Accuracy	0.98	0.21	**1**	0.6	0. 69	1
AUC	0.998	0.301	0.999	0.872	0.889	1
P_{R100}	0.98	0.214	1	0.625	0.704	1
	Gardens Point (day-to-day) Dataset					
VPR Technique	NetVLAD	HOG	CALC	AMOSNet	HybridNet	ConvSequential-SLAM
K	1	1	10	1	1	dynamic
t_e (sec)	0.77	0.0043	0.27	0.36	0.36	-
t_m (sec)	0.199	2.18	0.622	0.773	0.78	-
t_{VPR} (sec)	0.969	2.184	**0.892**	1.133	1.14	-
Accuracy	**0.955**	0.315	0.75	0.76	0.81	1
AUC	0.959	0.431	0.899	0.907	0.933	1
P_{R100}	0.955	0.316	0.748	0.779	0.814	1

mentioned above.

HOG has the lowest encoding time t_e of all VPR techniques presented. However, due to its increased matching time t_m, it is unable to achieve a sequence length of $\mathbf{K} > 1$ in less t_{VPR} than NetVLAD. On the other hand, CALC has an overall low t_{VPR}, thus being able to compute a longer sequence length than every other technique. We show in Table 4.3 that, on Campus Loop dataset, CALC is able to achieve better performance than NetVLAD, in less t_{VPR}. However, due to the low single-frame matching performance of CALC on datasets such as Gardens Point (day-to-night) and Nordland (as shown in Fig. 4.4), a much longer sequence length **K** than the one obtained in the given t_{VPR} would have been required to achieve the same or better levels of performance as other CNN-based VPR techniques such as NetVLAD, AMOSNet or HybridNet (refer to Table 4.4).

This experiment concludes that, on Campus Loop dataset, CALC with a sequence length

Table 4.4: Given the t_{VPR} of the best performing single-frame-based VPR technique, we show the maximum sequence length that can be reached by the sequence-based implementation of the remaining VPR techniques, on Gardens Point (day-to-night) and Nordland datasets.

	Gardens Point (day-to-night) Dataset					
VPR Technique	NetVLAD	HOG	CALC	AMOSNet	HybridNet	ConvSequential-SLAM
K	1	1	10	1	1	dynamic
t_e (sec)	0.77	0.0043	0.27	0.36	0.36	-
t_m (sec)	0.223	2.13	0.632	0.768	0.779	-
t_{VPR} (sec)	0.993	2.134	**0.902**	1.128	1.139	-
Accuracy	**0.565**	0.205	0.355	0.475	0.45	0.607
AUC	0.698	0.294	0.623	0.571	0.595	0.8
P_{R100}	0.585	0.214	0.357	0.477	0.456	0.649
	Nordland Dataset					
VPR Technique	NetVLAD	HOG	CALC	AMOSNet	HybridNet	ConvSequential-SLAM
K	1	1	13	1	1	dynamic
t_e (sec)	0.77	0.0043	0.351	0.36	0.36	-
t_m (sec)	0.155	1.62	0.563	0.536	0.613	-
t_{VPR} (sec)	0.925	1.624	0.914	**0.896**	0.973	-
Accuracy	**0.412**	0.023	0.143	0.162	0.186	0.537
AUC	0.733	0.036	0.39	0.132	0.214	0.6
P_{R100}	0.42	0.04	0.143	0.163	0.187	0.544

of 16 images can achieve better and faster place matching performance within the computational budget represented by the t_{VPR} of the single-frame-based implementation of NetVLAD ($\mathbf{K} = 1$). For this reason, we propose that the sequence-based implementation of CALC (with a sequence length of $\mathbf{K} = 16$ images) is selected as an alternative to the single-based implementation of NetVLAD on this dataset, as presented in our experiment.

4.5 Summary

To bridge the gap caused by a lack of systematic study on sequence-based filtering for visual route-based navigation, this chapter has conducted an in-depth investigation on the benefits and trade-offs of sequence-based filtering on top of single-frame-based VPR methods. This analysis is performed on four public sequential VPR datasets, that pose difficulties in place

matching (appearance changes, viewpoint variations etc.), using a variety of widely used performance metrics such as PCU. Sequential filtering is introduced into a number of contemporary single-frame-based VPR methods in order to present the findings. The results show the effects of various sequence lengths on performance boost and suitable combinations of different VPR techniques and sequence lengths are determined. Moreover, we take into consideration the computational effects of sequential-based filtering, for enabling the best place matching performance in different scenarios. Therefore, the focus of this chapter is to show that VPR can be performed more efficiently by pairing lightweight single-frame-based techniques with sequence-based filtering, thus being capable of outperforming more complex VPR descriptors.

Chapter 5

Data-Efficient VPR Using Extremely JPEG-Compressed Images

In contrast with chapters 3 and 4 whose focus was directed towards sequence-based filtering, this chapter shifts its attention towards an unexplored avenue for VPR research, namely image compression. JPEG is a widely used image compression standard that is capable of significantly reducing the size of an image at the cost of image clarity. For applications where several robotic platforms are simultaneously deployed, the visual data gathered must be transmitted remotely between each robot. Hence, JPEG compression can be employed to drastically reduce the amount of data transmitted over a communication channel, as working with limited bandwidth for VPR can be proven to be a challenging task. However, the effects of JPEG compression on the performance of current VPR techniques have not been previously studied. For this reason, this chapter presents an in-depth study of JPEG compression in VPR related scenarios. We use a selection of well-established VPR techniques on well-established benchmark datasets with various amounts of compression applied. We show that by introducing compression, the VPR performance is drastically reduced, especially in the higher spectrum of compression. Moreover, this chapter demonstrates how fine-tuning a CNN can be utilised as an optimisation method for JPEG compressed data to perform more consistently with the image transformations detected in extremely JPEG compressed images.

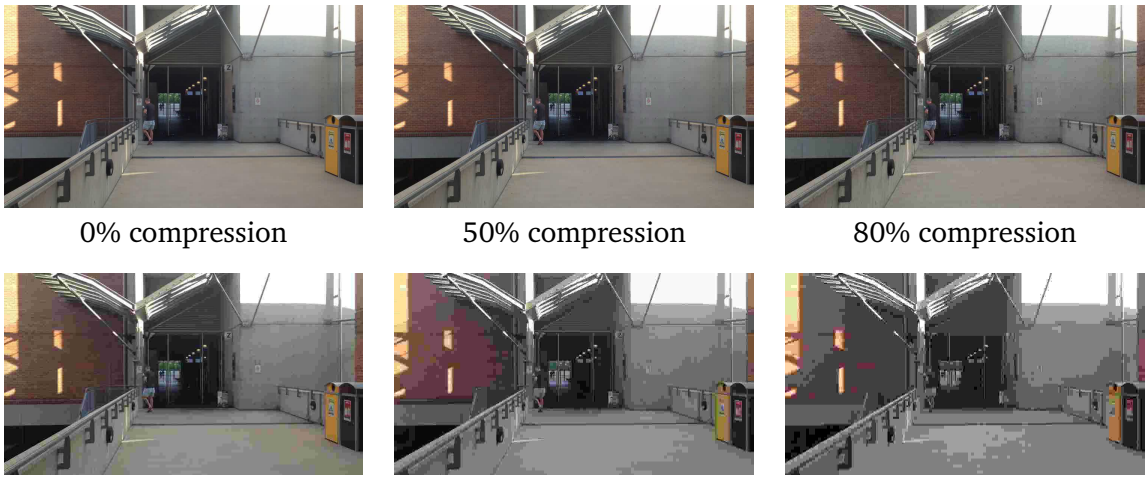


Figure 5.1: The same image taken from the Gardens Point *day left* dataset with different compression percentages applied.

5.1 Introduction

In this chapter, we propose to use highly compressed JPEG images to reduce the amount of data transmitted in decentralised VPR contexts, as identified in sub-section 2.6 of the literature review. ISO/IEC-ITU JPEG is one of the most widely used compression standards employed to facilitate significant data storage and transmission reduction [143]. The high compression ratios enabled by JPEG compared to other standard techniques [144, 145] make it attractive for distributed VPR applications. However, JPEG is designed to have a minimal impact on the human perception system [146]. It is uncertain how the performance of various VPR techniques is affected throughout the compression spectrum. To the best of our knowledge, the study of JPEG for VPR application has been overlooked so far. To bridge this gap, this chapter proposes to tune CNNs that deal with highly compressed JPEG images to circumvent the limitation of existing techniques. In summary, our contributions are as follows:

- An assessment of several well-established VPR techniques under mild to extreme JPEG compression rate, as shown in Fig. 5.1. This analysis uses several datasets presenting illumination, viewpoint, and weather variations to cover some of the most common viewing conditions experienced by a robot in real-world decentralised applications, where the operating environment might present heterogeneous conditions in different places (see Fig. 5.2).

- We demonstrate how a fine-tuned CNN-based descriptor on highly JPEG compressed data can achieve higher and more consistent VPR performance than non-optimised VPR techniques.

5.2 Methodology

5.2.1 Image Compression

JPEG is a compression method that allows the user to select and adjust the amount of compression applied to an image. As JPEG is a lossy compression method, there is a trade-off between image clarity and image size. By applying an increased amount of compression to any given image (e.g. above 90%), artifacts are introduced in the resulting compressed image, which will inevitably lead to image alteration as seen in Fig. 5.1.

The JPEG compression process can be broken down into three main steps. Firstly, the data in a given image is divided into the color and luminance components. As the human perception system is better suited to perceive intensity rather than color information, the latter can be subsampled to reduce the amount of data whilst maintaining the image visually unchanged to the user. Secondly, the data subsampled from the color component is divided into 8x8 pixel blocks. On each block, the Discrete Cosine Transform (DCT) is applied to describe the image content by the coefficients of the spatial frequencies for vertical and horizontal orientations, instead of pixel values. Finally, data quantization is performed, where the higher frequency coefficients are transformed to 0 first. Depending on the amount of compression selected by the user, the subsampling step may be skipped to achieve mild image compression. Conversely, to extremely compress an image, the subsampling step is turned on, and the quantisation matrix is selected so that most coefficients are set to 0.

The amount of compression applied to a given image is a parameter of the JPEG function [143], having values in range [0,99]. As the visual quality of the image is not compromised in the lower spectrum of JPEG compression, a lower value (e.g. 50%) should be selected to achieve mild image compression. Conversely, for an extremely JPEG compressed image, a high value (e.g. 97%) should be assigned to the compression parameter.

5.2.2 Place Matching

The place matching approach employed in this chapter has been previously detailed in chapter 4. Refer to sub-section 4.2.1 for details related to single-based image matching.

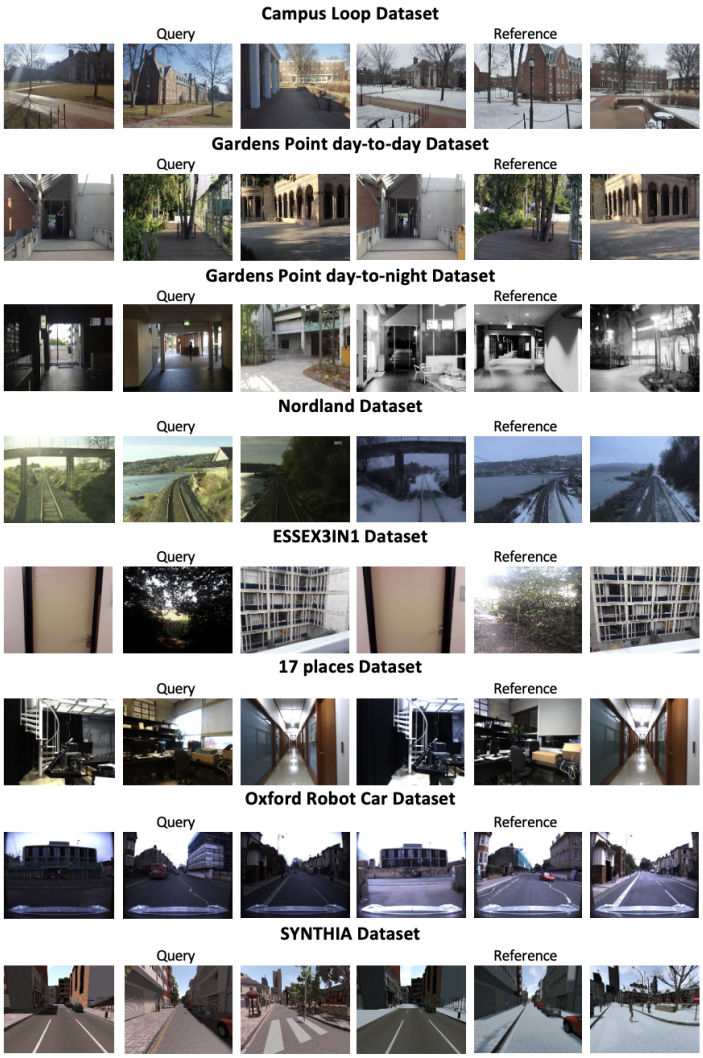


Figure 5.2: A selection of uncompressed query images and their corresponding reference images taken from each dataset.

5.2.3 Performance Metric

In this chapter, VPR performance is evaluated using the percentage of correctly matched images as discussed in sub-section 2.7.2, utilising equation (2.5).

5.3 Experimental Setup

This section presents the experimental setup for our work. We present the datasets used together with the VPR techniques employed for testing the effects of JPEG compression for VPR.

Table 5.1: The size of each dataset in Megabytes (MB) with different JPEG compression ratios applied.

Dataset	Compression Applied					
	0%	**50%**	**80%**	**90%**	**95%**	**97%**
17 places	81.8	18.9	10.8	8.3	6.6	5.9
Nordland	235.1	26	13.7	8.3	5.4	4.4
Campus Loop	46.8	9	4.6	2.6	1.5	1
Gardens Point (day-to-day)	54.8	17.6	8.4	5.2	3.1	2.3
Gardens Point (day-to-night)	44.9	15.2	7	4.2	2.5	1.8
Oxford Robot Car	185.7	28.8	15.6	9.7	6.2	4.7
ESSEX3IN1	1100	191.2	100	56.8	29.9	19.6
SYNTHIA	207.5	33.6	15.9	8.5	4.8	3.6

5.3.1 Test Datasets

The test data consists of eight datasets designed for VPR applications (discussed in-depth in sub-section 2.7.1). Fig. 5.2 shows a query-reference pair for each of them. The datasets are as follows: Campus Loop [81], Gardens Point (GP) [39] day-to-day and day-to-night, Nordland [78], ESSEX3IN1 [6], 17 places [129], SYNTHIA [128] and Oxford Robot Car (ORCD) [12]. Table 5.1 presents the size of each dataset (in Megabytes) for different levels of JPEG compression.

5.3.2 VPR Techniques

In this work, six well-established VPR techniques are used to show the effects of JPEG compression in VPR scenarios. These techniques are as follows: HOG [65], CALC [81], HybridNet [13], NetVLAD [74], CoHOG [67] and AlexNet [39]. All VPR techniques are used as they are presented by their authors with no additional changes being made to neither technique. For a fair comparison with our model, the results for AlexNet have been generated utilising the *fc6* layer [39].

As mentioned in section 5.2.1, JPEG compression introduces artifacts while decreasing the quality of the image. As a result, JPEG compression introduces appearance changes in an image, rather than viewpoint changes. The selection of VPR techniques employed in this

work can be divided into two main categories: VPR techniques that are robust to viewpoint changes (such as NetVLAD and CoHOG) and techniques that are optimised for appearance changes (such as HybridNet and AlexNet). Thus, we can identify which approach can be easily adapted to deal with extreme JPEG compression rates.

5.4 Results and Analysis

In sub-section 5.4.1, we present the effects of JPEG compression on the performance of several VPR techniques. We discuss the performance of each technique for several levels of compression in terms of accuracy. Furthermore, in sub-section 5.4.2, the details of our JPEG optimised CNN are provided, whilst also presenting a comparison between our model trained on compressed data and other VPR techniques. In sub-section 5.4.3 we present the place matching performance of our model in scenarios where the query and map images may have different levels of JPEG compression applied (non-uniform compression).

5.4.1 Place Matching Performance

By increasing the compression percentage on each dataset, we generally obtain lower results. This can be seen in Fig. 5.3, where the accuracy (y-axis) generally decreases with the increase in compression rate. This descending trend in performance is expected due to the fact that an increase in JPEG compression would conclude in a drastic change within the image structure (as observed in Fig. 5.1).

The results presented in Fig. 5.3 show that the amount of compression applied to each dataset has a direct effect on the place matching performance. However, each technique is affected differently by image compression. A possible explanation for this decrease in place matching performance can be related to the inability of current VPR techniques to cope with the extreme changes that emerge from including image compression besides the already existing challenges in VPR (viewpoint, illumination, seasonal variations etc.). In particular, we observed that JPEG compression affects more those methods designed to deal with viewpoint changes such as NetVLAD and CoHOG. On the contrary, VPR descriptors that are designed to handle appearance changes present higher tolerance to JPEG compression. The details of our analysis are presented below.

On datasets including illumination variation, such as Gardens Point day-to-night and SYNTHIA, techniques such as NetVLAD, CoHOG and HOG lose significant performance through-

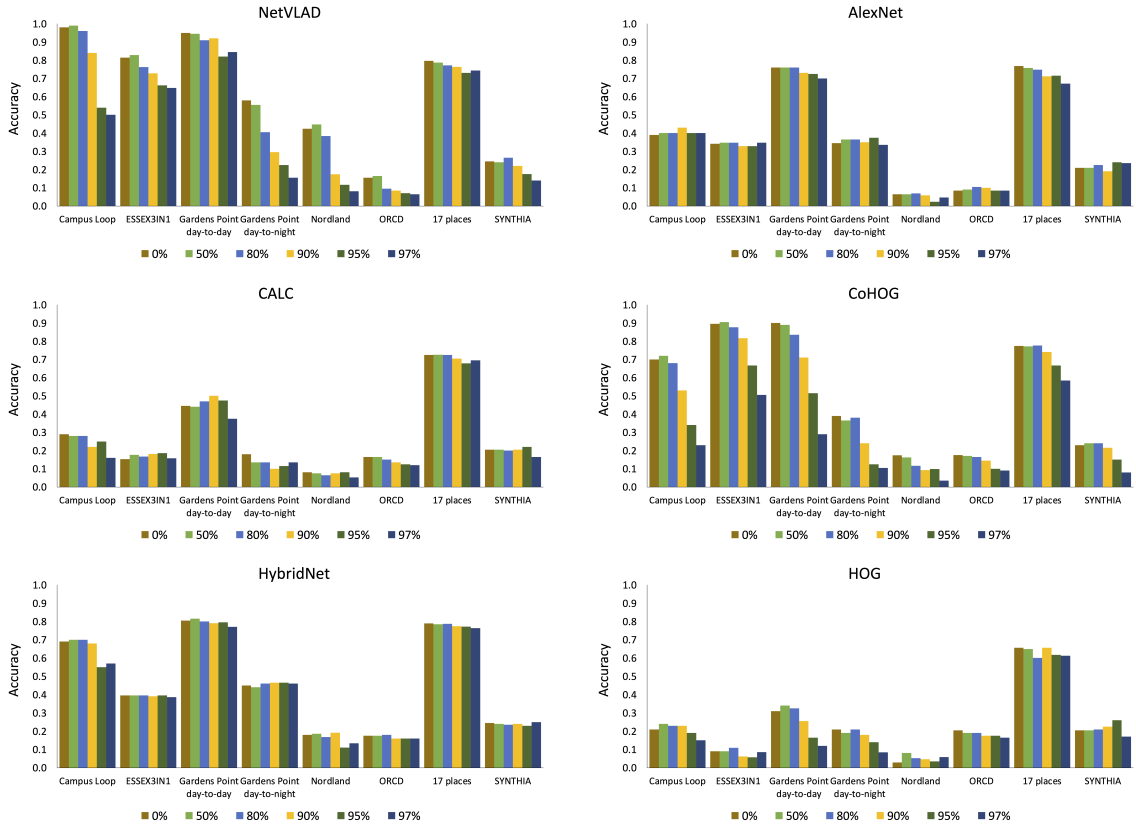


Figure 5.3: The accuracy of all VPR techniques on each dataset with different levels of JPEG compression applied is presented here.

out the JPEG compression spectrum. The most affected VPR technique on 17 places dataset is CoHOG, where the application of JPEG compression translates to a prominent decrease in performance, as shown in Fig. 5.3. However, the results for SYNTHIA show that it is slightly more stable than Gardens Point. As SYNTHIA is a synthetic dataset, it is less information rich than a real-word dataset such as Gardens Point (refer to Fig. 5.2). The application of JPEG compression on the SYNTHIA dataset does not alter significantly the image content from the perspective of VPR. This conclusion is supported by Fig. 5.4 that shows the average entropy [34, 67] (on the y-axis) in each query dataset resulting from applying different levels of JPEG compression. The reduction in entropy on SYNTHIA is much smaller when compared to other datasets tested. It is worth mentioning that for both Gardens Point day-to-day and day-to-night datasets we use *day left* images as query images, therefore we only provide the entropy results once in Fig. 5.4.

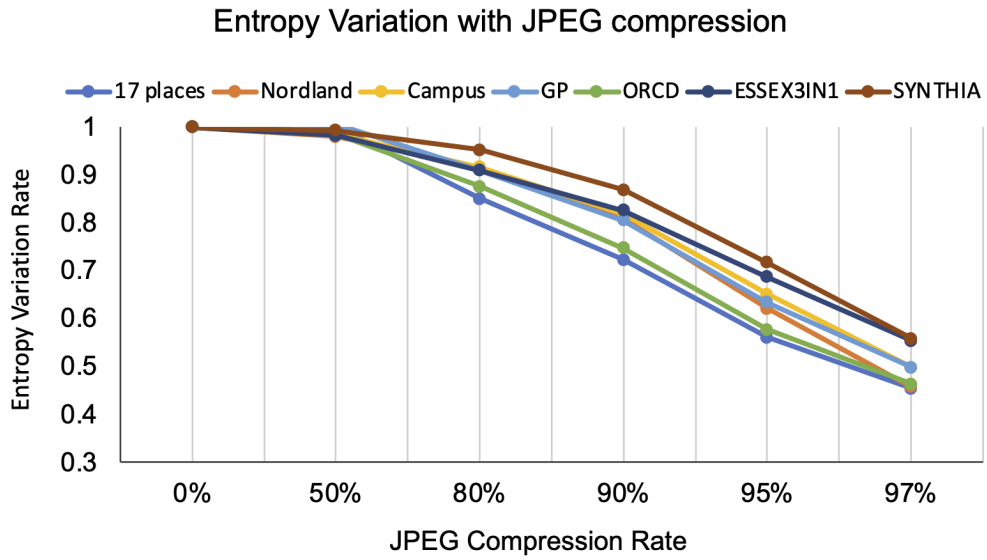


Figure 5.4: Average entropy in query images with different compression ratios applied to each dataset.

5.4.2 JPEG Optimised CNN

Model Design

In an effort to achieve more consistent place matching performance for different levels of JPEG compressed data, we fine-tuned a neural-network based VPR technique specifically for image compression. The neural network has the same structure as AlexNet [39], and has been trained on the Places365 dataset [147], which contains approximately 1.8 million images from 365 scene categories. Then, this neural network has been fine-tuned using 97% JPEG compressed versions of the images taken from the above mentioned dataset. We have specifically selected 97% JPEG compression rate as it provides the best trade-off between performance and stability. The resulting model, entitled *97*, achieves great stability and consistent performance in the higher compression spectrum, on a variety of environments and viewing conditions. Our model achieves a considerable improvement in place matching performance on JPEG compressed data over AlexNet, while at the same time being capable of matching and even outperforming the deeper HybridNet at high compression ratios.

Model Stability and Performance

We have previously mentioned in section 5.3.2 that each VPR technique has a different performance depending on the type and state of the perceived environment. Moreover, due to

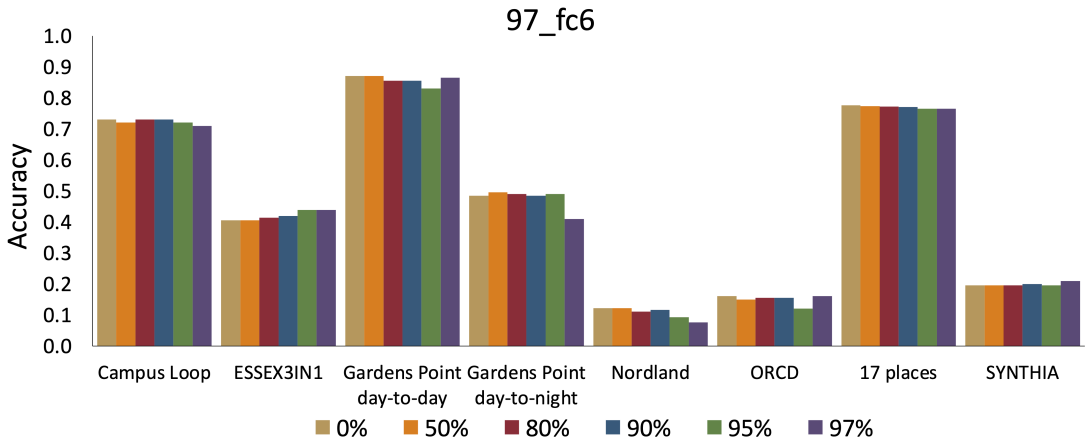


Figure 5.5: The accuracy of our model on all 8 datasets is enclosed here.

the presence of artifacts in a low-quality image as a result of utilising high levels of JPEG compression, the place matching performance of each VPR technique is reduced, as discussed in sub-section 5.4.1. To successfully perform VPR tasks in real world applications, it is fundamental to determine the best technique with regards to the above-mentioned environmental variables. Thus, in this sub-section, we present a comparison between the place matching performance of our model and that of other VPR techniques, throughout the entire spectrum of JPEG compression.

Fig. 5.3 shows that JPEG compression has drastic effects on the performance of most VPR techniques, especially when using a significant amount of compression (e.g. 97%). As there is no universal model that achieves the highest VPR performance on all tested datasets, we present in Fig. 5.5 the accuracy of one of the best performing models, with different amounts of compression applied to each dataset. We have selected the features from the *fc6* layer as they achieve the best performance on average.

The performance of our *97_fc6* model is shown in Fig. 5.5. The VPR accuracy is highly stable across different amounts of JPEG compression. Fig. 5.6 compares the average VPR performance between our model and the other VPR techniques, across all tested datasets. The results presented in Fig. 5.6 show that our model has more consistent performance on compressed data and tends to have a steadier decrease in performance throughout the compression spectrum. As previously mentioned, there is a trade-off in the performance and stability of each VPR technique with respect to the amount of image compression applied. While outperformed on low compression ratios by NetVLAD, HybridNet and CoHOG, our JPEG optimised model can better operate at the highest compression ratios. This is highlighted in Fig.

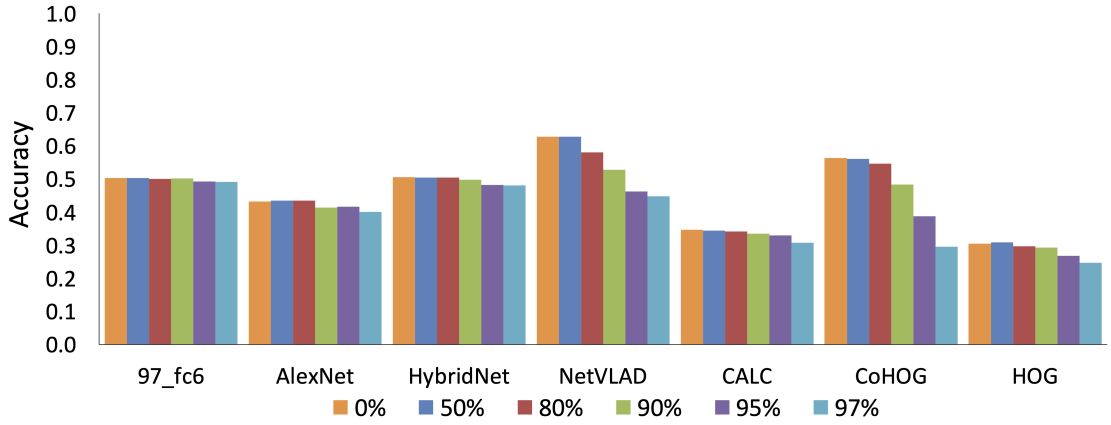


Figure 5.6: The average accuracy of our model in comparison with other VPR techniques on the combined datasets, for each level of JPEG compression applied.

5.3 on the Campus Loop dataset, where NetVLAD achieves the highest place matching performance for compression levels of up to 90%. However, when compression reaches 97%, our model should be used instead as it achieves higher place matching performance as reported in Fig. 5.5. This observation also applies to the Gardens Point day-to-day dataset. Moreover, on Gardens Point day-to-night dataset, our model outperforms every technique for compression levels of above 80%. Our *97_fc6* model outperforms AlexNet on all JPEG compressed datasets, except for SYNTHIA as seen in Fig. 5.3. However, on the 97% compressed versions of Nordland, Oxford Robot Car and ESSEX3IN1 datasets, our model is outperformed by some VPR techniques presented in Fig. 5.3. In these cases, the technique that achieves the highest VPR performance should be utilised instead.

5.4.3 Non-Uniform JPEG Compressed Datasets

In some practical cases, the visual data transmitted by an agent might be subject to bandwidth limitations, raising the need to use highly compressed images. Hence, the query and stored images (e.g. the map) may have different JPEG compression levels applied (non-uniform compression). For this reason, an analysis on the effects of non-uniform JPEG compression on the performance of our model and other VPR techniques is presented below.

Fig. 5.7 presents the average performance of every VPR technique in scenarios where all datasets are non-uniformly compressed. We only included the most significant results, those of the extremes of the JPEG compression spectrum. Apart from being stable on highly JPEG compressed images as discussed in sub-section 5.4.2, our model also has consistent perform-

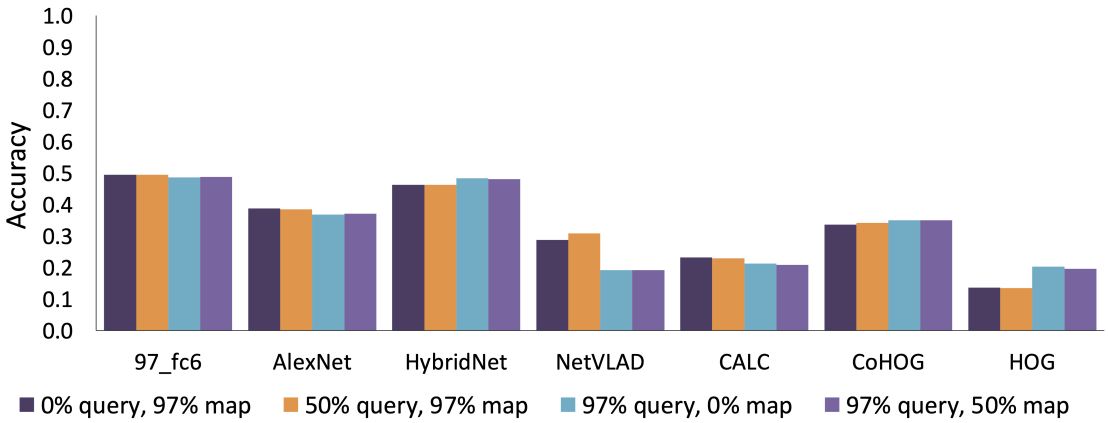


Figure 5.7: The average place matching performance of our model in comparison with the other VPR techniques presented, in scenarios where the amount of JPEG compression applied to query and reference images greatly differs.

ance on datasets where the amounts of JPEG compression applied to the query and reference images are in the opposite spectrum. Our *97_fc6* model outperforms AlexNet and achieves slightly better overall performance than HybridNet on non-uniform JPEG compressed data.

Fig. 5.8 presents the detailed results on Campus Loop and SYNTHIA datasets with non-uniform JPEG compression applied. On the Campus Loop dataset, our model achieves the highest VPR performance, outperforming every VPR technique tested. In sub-section 5.4.1 we have shown that SYNTHIA is more stable under JPEG compression (due to its synthetic nature) in contrast to other datasets taken from real-world environments. This observation is also emphasized in Fig. 5.8, which shows that the performance of most VPR techniques on the SYNTHIA dataset is not drastically affected in the presence on non-uniform JPEG compression, especially when compared with the results presented in Fig. 5.3.

The results presented in Fig. 5.7 and Fig. 5.8 show that our model is more tolerant to non-uniform compression than any of the other VPR techniques tested. Moreover, the results presented throughout this chapter emphasize the exceptional performance stability achieved by our neural-network on both uniform and non-uniform JPEG compressed data.

5.5 Summary

This chapter conducts an in-depth study on the effects of JPEG compression in VPR. We use a selection of well-established VPR techniques on a variety of JPEG compressed VPR datasets

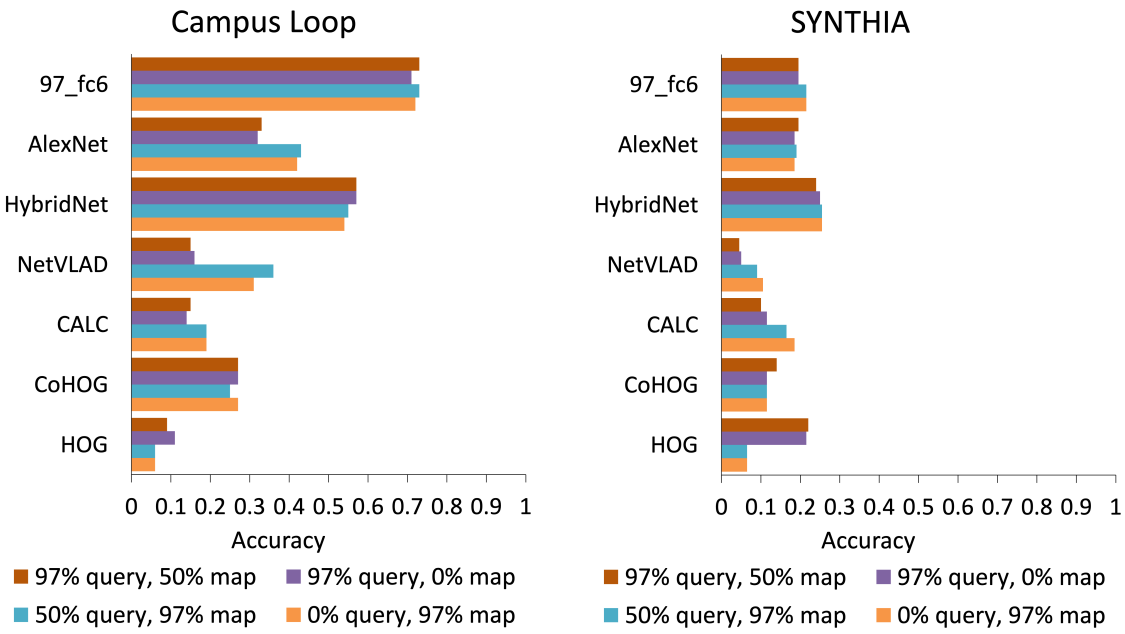


Figure 5.8: The accuracy of our model in comparison with each VPR technique on non-uniform JPEG compressed versions of the Campus Loop and SYNTHIA datasets.

to present our findings. Our experiments show that techniques which are designed to deal with appearance changes present higher tolerance to JPEG compression in comparison with techniques that are designed to handle viewpoint variations. In an attempt to achieve more stable VPR performance when using JPEG compressed data, we demonstrate how fine-tuning can optimise a CNN descriptor to handle highly compressed images. The results show that our model is more consistent on both uniform and non-uniform JPEG compressed data than any other VPR technique presented in this work.

Chapter 6

Data-Efficient Sequence-Based VPR for JPEG-Compressed Imagery

As previously mentioned in chapter 5, JPEG is not designed with VPR applications in mind. Thus, it introduces a new level of complexity on top of the already existing challenges in VPR, translating to reduced place matching performance. In this chapter, we incorporate the sequence-based filtering schema utilised in both chapter 3 and chapter 4, in a number of well-established, learnt and non-learnt VPR techniques to overcome the performance loss resulted from introducing high levels of JPEG compression. The sequence length that enables 100% VPR performance is reported, whilst also providing an analysis of the amount of data required for each VPR technique to perform the transfer on the entire spectrum of JPEG compression. Moreover, the time required by each VPR technique to perform place matching is investigated on both uniformly and non-uniformly JPEG compressed data. The results show that it is beneficial to use a highly compressed JPEG dataset with an increased sequence length, as similar levels of VPR performance are reported at a significantly reduced bandwidth. The results presented in this chapter also emphasize that there is a trade-off between the amount of data transferred and the total time required to perform VPR. Our experiments also suggest that is often favourable to compress the query images to the same quality of the map, as more efficient place matching can be performed. The experiments are conducted on several VPR datasets, under mild to extreme JPEG compression.

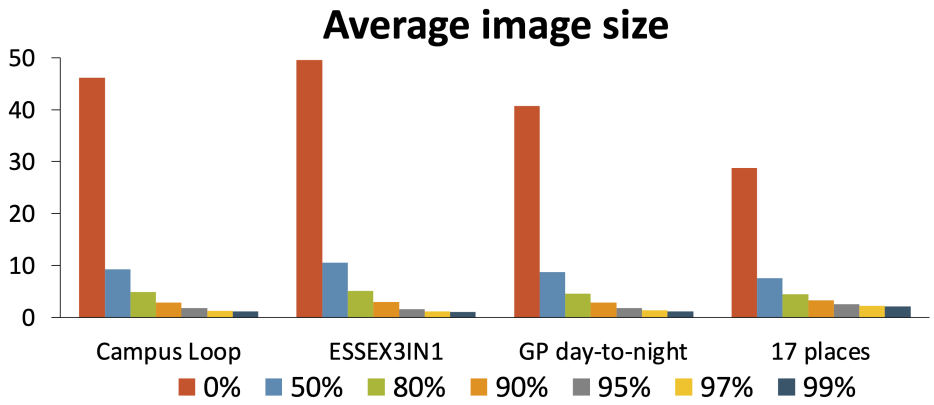


Figure 6.1: The average image size in Kilobytes (KB) taken from each dataset with multiple JPEG compression ratios applied.

6.1 Introduction

We have previously shown in chapter 5 that the VPR performance is drastically reduced especially when working with highly JPEG compressed images. Moreover, we have shown in chapter 4 that by introducing sequence-based filtering on top of single-frame based VPR techniques, their place matching performance is greatly improved. In this chapter, to compensate for the performance degradation resulted from utilising high ratios of JPEG compression, we propose to introduce the sequence-based filtering schema devised in chapter 3 and later utilised in chapter 4, in several well-established VPR techniques. We analyse the total time required to perform VPR to determine whether it is more efficient to compress the query images at the same quality as the map. In summary, our contributions are as follows:

- The application of sequence-based filtering is investigated on highly JPEG compressed data. An analysis of the sequence length that enables perfect place matching performance and the amount of data required to transmit for each VPR technique is provided. This study is performed on several datasets containing illumination, viewpoint and seasonal variations, accurately depicting the most widely encountered changes in the environment.
- The time required for each VPR technique to perform place matching is investigated throughout the entire spectrum of JPEG compression, in scenarios where the datasets are uniformly and non-uniformly JPEG compressed. The analysis suggests that both the query and map images should be compressed at the same ratio, as it facilitates more efficient VPR at reduced bandwidth.

6.2 Methodology

6.2.1 JPEG Compression

Refer to chapter 5 sub-section 5.2.1 for details related to the JPEG compression process. Fig. 6.1 illustrates that with higher levels of JPEG compression, the size of an image is drastically reduced. For this reason, JPEG compression can benefit decentralised VPR applications as less visual data is required to be transmitted. It is important to note that JPEG compression only affects the image's quality and file size.

6.2.2 Implementation of Sequence-Based Filtering

The details of the implementation of both the single-image-based and sequence-based filtering have been discussed at length in chapter 4 and are not provided here to avoid redundancy. Refer to sub-section 4.2.1 for the single-image-based implementation and sub-section 4.2.2 for the sequence-based filtering implementation.

6.3 Experimental Setup

6.3.1 VPR Techniques

A selection of four well-established, learnt and non-learnt VPR techniques have been employed in this work including: NetVLAD [74], HybridNet [13], RegionVLAD [83] and HOG [65]. These techniques have been selected as they span over a broad spectrum ranging from descriptors that are designed for seasonal and illumination variations such as HybridNet and descriptors that achieve viewpoint tolerances such as NetVLAD. The sequence-based filtering schema proposed in chapter 3 has been included in every VPR technique mentioned above as it can be generalized for every method, *viz.* it is agnostic to the underlying single-frame technique. For this reason, this particular matching schema is utilised as it enables a fair comparison of each sequence-based VPR technique on highly JPEG compressed data.

6.3.2 Test Datasets

This chapter employs a total of four datasets (presented in sub-section 2.7.1) widely used by the VPR community, which present different scenarios where the environment is affected by illumination, viewpoint and/or seasonal variations. These are as follows: Campus Loop [81],

ESSEX3IN1 [6], Gardens Point (GP) [39] day-to-night and 17 places [129]. For the purpose of this work, we utilise 420 query (*day_vme1*) and 420 reference images (*night_vme1*), taken from the Arena, AshRoom and Corridor locations of the 17 places dataset.

To enable an image size comparison, the datasets have been resized to 224x224 pixels for the CNNs and 256x256 pixels (the closest power of 2 $-$ 2^8x2^8) for HOG. Moreover, resizing the datasets also helps in comparing the time required to perform VPR for HOG with the rest of the methods, as similar image resolution will be utilised by all techniques. Following the resizing process, multiple JPEG compression ratios have been applied. Fig. 6.1 presents the average image size - in Kilobytes (KB) - for each of the four datasets utilised, throughout the entire spectrum of JPEG compression. We only plot the image size for the datasets resized to 256x256 pixels in Fig. 6.1, as there is no major difference in size between the two types of resized datasets. It can be seen that significant size reduction can be achieved, especially when applying a high JPEG compression ratio.

6.3.3 Performance Metric

Similarly to chapter 5, the VPR performance of each technique employed in this work is evaluated using the percentage of correctly matched images, utilising equation (2.5).

6.4 Results and Analysis

This section presents the analysis of each sequence-based VPR technique on the employed test data. The sequence length that enables perfect place matching performance throughout the entire spectrum of JPEG compression is reported, utilising both uniformly (same JPEG compression ratio is utilised for both query and map images) and non-uniformly (compression ratio differs from query to map images) compressed datasets. An analysis on the amount of data transferred and the time required to perform VPR for each technique utilised is also included in this section. All experiments have been conducted on a PC equipped with an Intel Core i7-4790k CPU.

6.4.1 Sequence Length Impact on VPR

JPEG compression reduces a VPR descriptor's ability of performing successful place matching, especially in the higher spectrum of compression, as discussed in chapter 5. However, VPR techniques designed to handle appearance changes $-$ such as HybridNet $-$ are more toler-

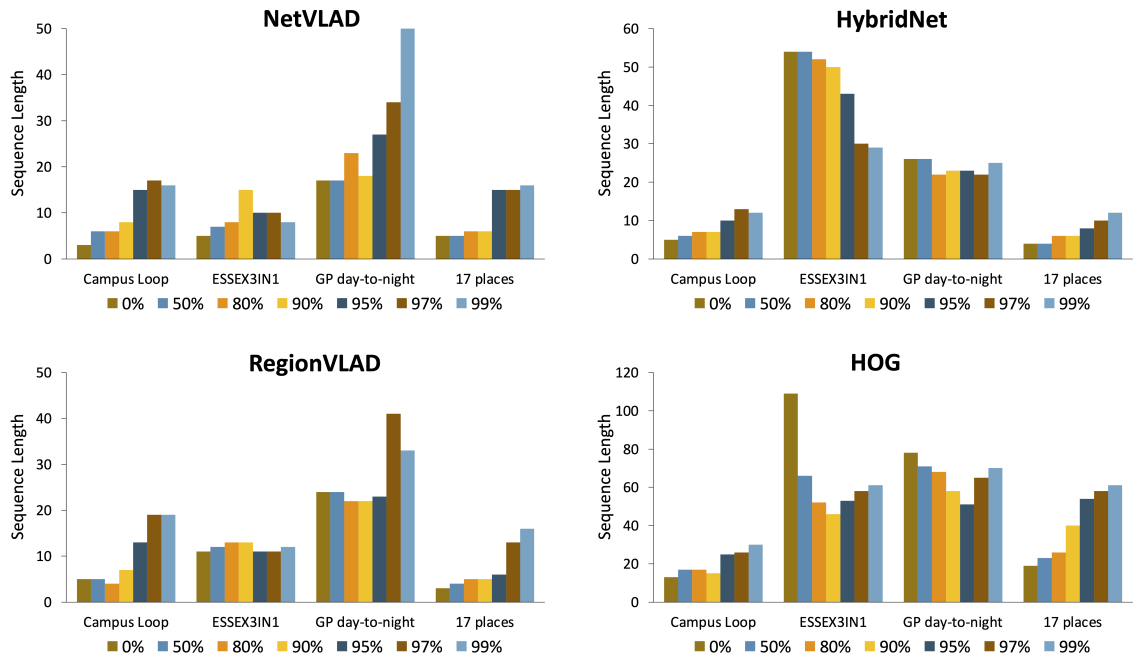


Figure 6.2: The sequence length required for each VPR technique to reach maximum accuracy for each JPEG compression ratio is enclosed here.

ant to high levels of JPEG compression in contrast with descriptors designed for viewpoint changes – such as NetVLAD – which are prone to severe loss in accuracy. To overcome the extreme loss in VPR performance resulted from introducing JPEG compression, sequence-based filtering is introduced in several VPR techniques (presented in sub-section 6.3.1). The details of our analysis are presented below.

6.4.2 Data Requirements for 100% accurate VPR

Fig. 6.2 presents the sequence length **K** that facilitates each VPR technique to achieve perfect place matching performance (100% accuracy), throughout the entire spectrum of JPEG compression. As utilising a higher level of JPEG compression usually yields a lower VPR performance, the sequence length **K** required to achieve maximum VPR accuracy is increased. This observation is highlighted in Fig. 6.2 on Gardens Point day-to-night, where NetVLAD requires a sequence length **K** three times higher on the 99% JPEG compressed dataset, in comparison with the uncompressed dataset. In comparison with the other VPR techniques tested, HybridNet is more tolerant to JPEG compression as the sequence length required to reach 100% accuracy is not greatly increased throughout the compression spectrum, as re-

Table 6.1: Descriptor sizes compared to the average image size of ESSEX3IN1 at several compression levels.

VPR Technique	**Descriptor Size [KB]**	**Descriptor-Image Size Ratio [%]**						
		0%	**50%**	**80%**	**90%**	**95%**	**97%**	**99%**
NetVLAD	16	309.3	65.5	32	18.3	9.9	7.3	6.68
HybridNet	30	165	34.9	17.1	9.7	5.3	3.9	3.56
RegionVLAD	384	12.9	2.7	1.3	0.8	0.4	0.3	0.27
HOG	31.6	156.6	33.2	16.2	9.2	5	3.7	3.38

ported in Fig. 6.2. On the ESSEX3IN1 dataset, higher levels of JPEG compression improve the performance of this technique. Hence, HybridNet benefits from JPEG compression on the ESSEX3IN1 dataset, achieving maximum accuracy at 99% JPEG compression utilising a lower sequence length **K** than on any other compression level employed.

Table 6.1 presents a comparison between the average image size taken from the ESSEX3IN1 dataset and the image descriptor size of each VPR technique. The results suggest that it would be beneficial transmitting the compressed image rather than the image descriptor. This is especially noticeable for RegionVLAD, whose descriptor size is considerably higher than any ratio of JPEG compression applied to a given image. For the remaining VPR techniques, above 50% JPEG compression, the average image size is less than their descriptor size as shown in Table 6.1.

In decentralised VPR applications where the visual data has to be shared between multiple robotic platforms, the amount of data transferred has to be carefully considered as to not hamper with the VPR process. At any given JPEG compression level, the amount of data transferred d by a VPR technique can be calculated as follows:

$$d = i_s \times K, \tag{6.1}$$

where i_s is the average image size (in Kilobytes) at the given JPEG compression level, and **K** represents the sequence length that enables the VPR technique to achieve maximum place matching performance. For each VPR technique employed in this study, Fig. 6.3 shows the amount of data d required to be transferred throughout the entire spectrum of JPEG compression. A common observation is that the amount of data required for transfer d decreases as the amount of JPEG increases. As the image's size i_s is greatly reduced with an increase in JPEG compression (as seen in Fig. 6.1), a longer sequence length **K** would not always result in a larger amount of data transfer d, as observed in Fig. 6.3. The results presented above

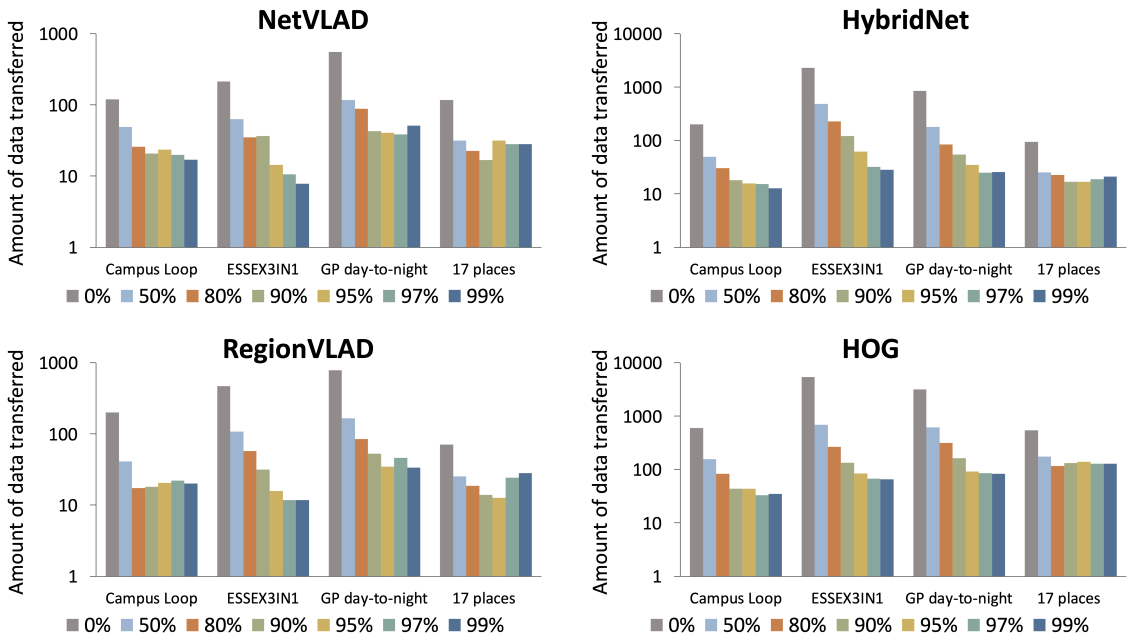


Figure 6.3: The amount of data transferred in Kilobytes (KB) for each VPR technique and JPEG compression ratio.

suggest that it is beneficial to use a higher JPEG compression rate paired with a longer sequence length **K**, as it allows the same levels of VPR performance to be achieved at decreased bandwidth requirements.

6.4.3 Non-Uniform Compression Ratios

To facilitate VPR applications where the limited bandwidth may disrupt the localisation process, the query and the map images may have different ratios of JPEG compression applied. Due to the discrepancy between the amount of JPEG compression applied to the query and reference images, the VPR performance may be drastically reduced. In Fig. 6.4, the sequence length required by each VPR technique to reach maximum accuracy on the non-uniformly JPEG compressed datasets is assessed. For each technique, we utilise the JPEG compression ratio that would result in the minimal amount of data transferred. The results presented in Fig. 6.4 show that each of the tested methods tend to perform worse on non-uniformly JPEG compressed data, requiring longer sequence lengths **K** to achieve maximum performance than on uniformly compressed data (refer to Fig. 6.2). Moreover, Fig. 6.4 also shows that by utilising a JPEG compressed query image and uncompressed map, most VPR techniques have a decrease in performance over the scenario where only the map is compressed.

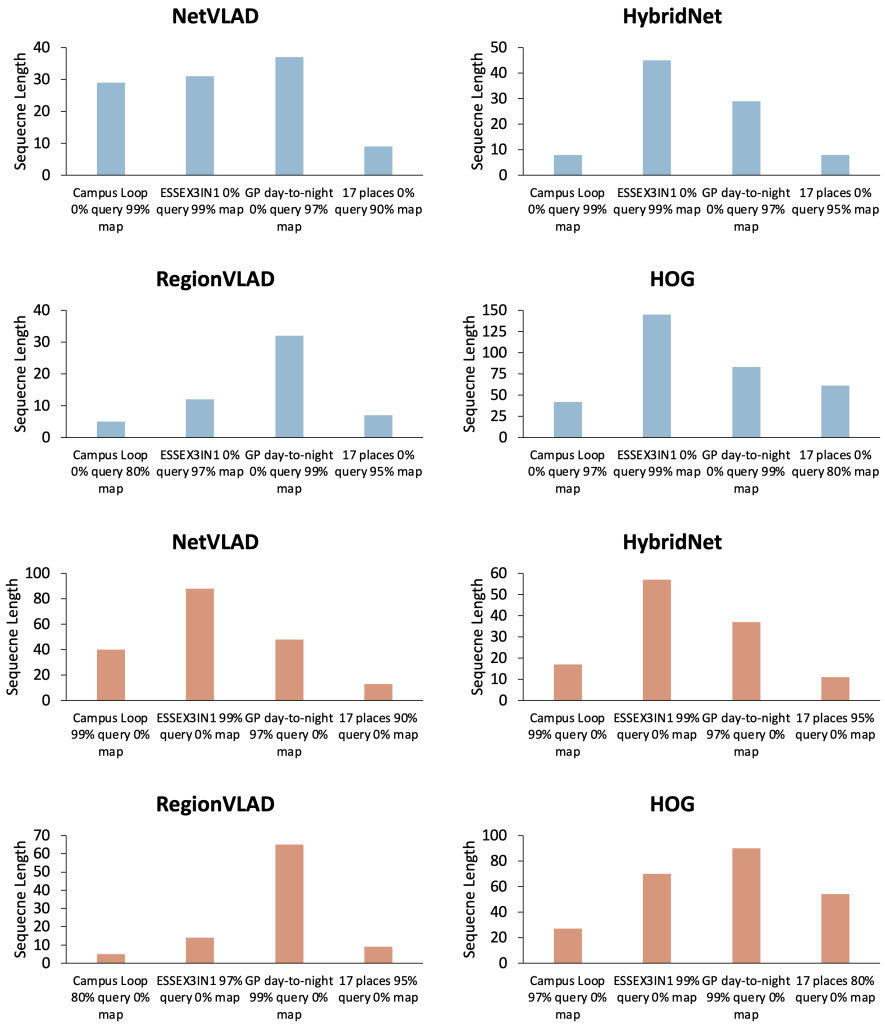


Figure 6.4: The value of K required to achieve maximum accuracy on non-uniformly JPEG compressed data.

6.4.4 Analysis on the Time Required to Perform VPR

Fig. 6.5 presents the VPR time of each technique for the sequence lengths **K** that are required to reach maximum place matching performance on each dataset and JPEG compression ratio (**K** values are presented in Fig. 6.2). In the case of a given sequence-based technique, the feature encoding time t'_e can be obtained by multiplying the feature encoding time of the single-image-based technique t_e (presented in Table 6.2) with the number of images in a sequence **K**, as follows:

$$t'_e = t_e \times K \tag{6.2}$$

To obtain the VPR time t_{VPR}, the matching time t_m is summed with the feature encoding

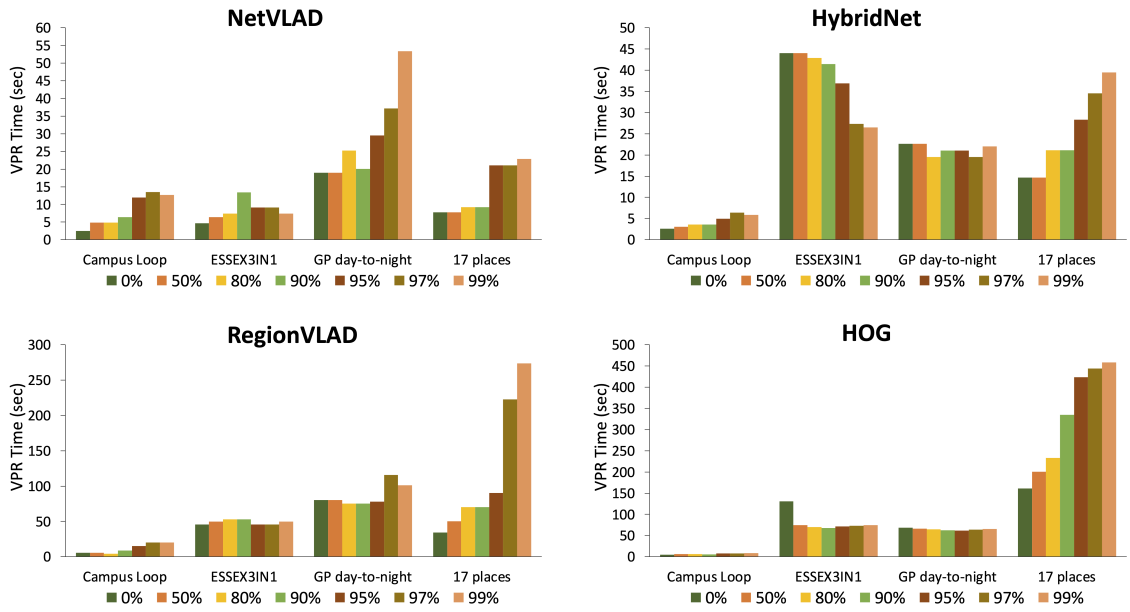


Figure 6.5: t_{VPR} for every VPR technique on each dataset and JPEG compression amount specified in Fig. 6.2

time t'_e as follows:

$$t_{VPR} = t_m + t'_e \tag{6.3}$$

As previously mentioned in sub-section 6.4.3 and shown in Fig. 6.4, the sequence length **K** required to achieve 100% accuracy when non-uniformly JPEG compressed datasets are employed is considerably higher in most cases than on uniformly compressed data. The results presented in both Fig. 6.5 and Fig. 6.6 clearly show that the sequence length **K** has a direct impact on t_{VPR}. As a result, for any given JPEG compression ratio, a VPR technique can drastically have its t_{VPR} increased if a longer sequence length **K** is employed.

Fig. 6.7 shows the time t_c required to apply JPEG compression to a given image for each of the four datasets tested. The time t'_c required to JPEG compress an entire sequence of images of length **K** can be computed using equation (6.4):

$$t'_c = t_c \times K \tag{6.4}$$

$$t_{total} = t_{VPR} + t'_c \tag{6.5}$$

In Table 6.3, Table 6.4, Table 6.5 and Table 6.6, a comparison is provided between the t_{VPR} in a scenario where each dataset employed is uniformly and non-uniformly JPEG compressed. The values in bold represent the shortest time required to perform VPR for each

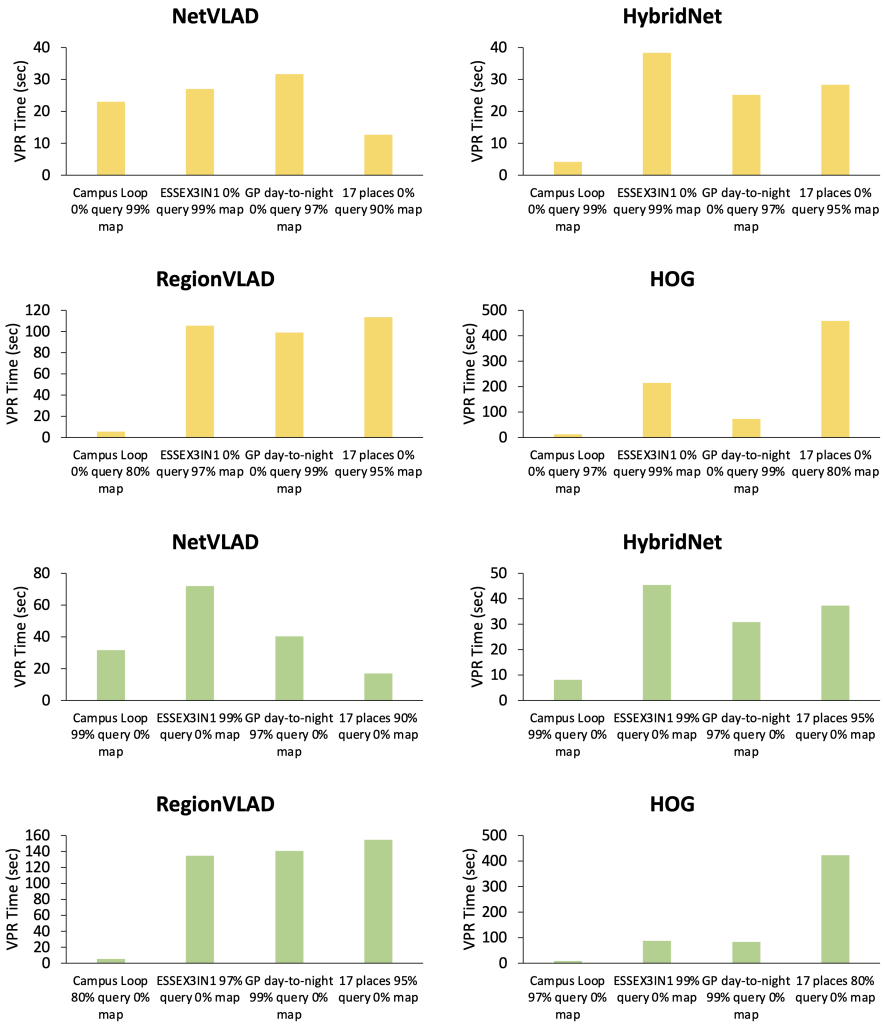


Figure 6.6: t_{VPR} of each VPR technique on non-uniformly JPEG compressed data specified in Fig. 6.4.

technique. These results show that the amount of time required to perform VPR can be drastically reduced if the sequence of query images is compressed to the same quality as the map. As JPEG compression is an extremely fast operation to perform, t_{total} (refer to equation (6.5)) is not considerably higher than t_{VPR}. The results presented show that t_{total} always benefits from using a shorter sequence length **K**, thus it is desirable to JPEG compress the query images to have the same quality as the map, as it facilitates more efficient place matching performance. However, we note some cases where the sequence length of a VPR technique is decreased on non-uniformly JPEG compressed data (refer to Fig. 6.2 and Fig. 6.4), more specifically HybridNet on Campus Loop (0% query and 99% map), RegionVLAD

Table 6.2: Feature encoding time of the single-image-based implementation of each VPR technique.

VPR Technique	t_e *(sec)*
NetVLAD	0.77
HybridNet	0.36
RegionVLAD	0.424
HOG	0.0043

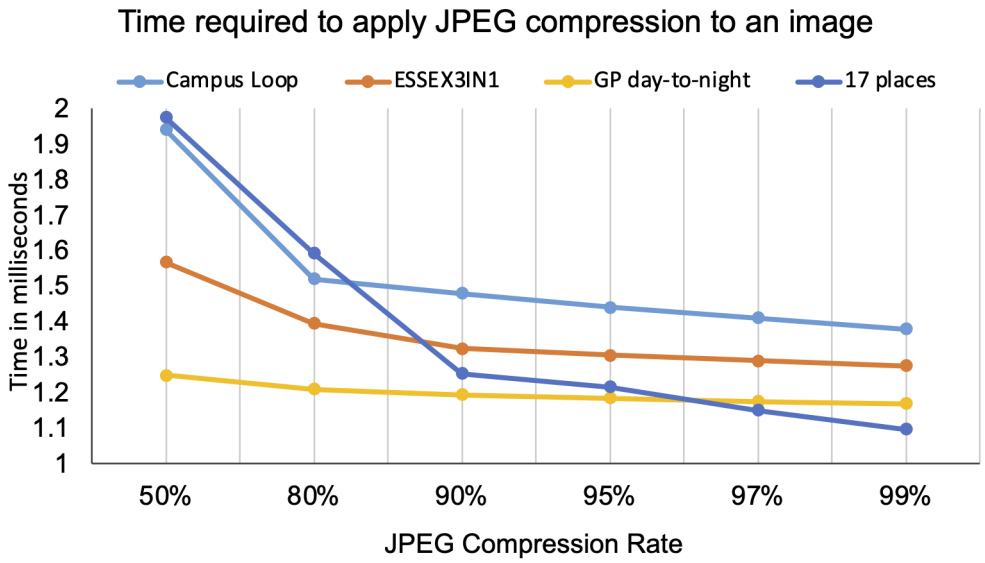


Figure 6.7: The average time t_c required to JPEG compress an image.

on Gardens Point day-to-night (0% query, 99% map) and HybridNet on 17 places (0% query, 95% map). In these scenarios, the sequence of query images should not be compressed to the same quality of the map, as it would increase the sequence length required to achieve maximum accuracy, while also increasing t_{VPR}.

Table 6.3: Total VPR time, in scenarios where the ESSEX3IN1 dataset is both uniformly and non-uniformly JPEG compressed.

	NetVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	8	1.205	6.16	7.365	0.0102	**7.375**
0% query, 99% map	31	3.1	23.87	26.97	-	26.97
99% query, 0% map	88	4.1	67.76	71.86	-	71.86
	HybridNet					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	29	16.05	10.44	26.49	0.037	**26.52**
0% query, 99% map	45	22.14	16.2	38.34	-	38.34
99% query, 0% map	57	24.95	20.52	45.47	-	45.47
	RegionVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
97% query, 97% map	11	40.96	4.664	45.624	0.0142	**45.638**
0% query, 97% map	30	92.78	12.72	105.5	-	105.5
97% query, 0% map	45	115.78	19.08	134.86	-	134.86
	HOG					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	61	73.94	0.262	74.2	0.0778	**74.277**
0% query, 99% map	145	214.09	0.623	214.71	-	214.71
99% query, 0% map	70	88.3	0.301	88.6	-	88.6

Table 6.4: Total VPR time, in scenarios where the Campus Loop dataset is both uniformly and non-uniformly JPEG compressed.

	NetVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	16	0.377	12.32	12.697	0.022	**12.719**
0% query, 99% map	29	0.68	22.33	23.01	-	23.01
99% query, 0% map	40	0.91	30.8	31.71	-	31.71
	HybridNet					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	12	1.582	4.32	5.902	0.0165	5.918
0% query, 99% map	8	1.24	2.88	4.12	-	**4.12**
99% query, 0% map	17	2.08	6.12	8.2	-	8.2
	RegionVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
80% query, 80% map	4	2.56	1.696	4.256	0.00608	**4.262**
0% query, 80% map	5	3.5	2.12	5.62	-	5.62
80% query, 0% map	5	3.5	2.12	5.62	-	5.62
	HOG					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
97% query, 97% map	26	7.563	0.1118	7.6748	0.0366	**7.711**
0% query, 97% map	42	11.7	0.1806	11.8806	-	11.8806
97% query, 0% map	27	7.63	0.1161	7.7461	-	7.7461

Table 6.5: Total VPR time, in scenarios where the GP day-to-night dataset is both uniformly and non-uniformly JPEG compressed.

	NetVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
97% query, 97% map	34	3.04	26.18	29.22	0.0399	**29.259**
0% query, 97% map	37	3.12	28.49	31.61	-	31.61
97% query, 0% map	48	3.37	36.96	40.33	-	40.33
	HybridNet					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
97% query, 97% map	22	11.64	7.92	19.56	0.0258	**19.585**
0% query, 97% map	29	14.7	10.44	25.14	-	25.14
97% query, 0% map	37	17.51	13.32	30.83	-	30.83
	RegionVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	33	87.41	13.992	101.402	0.0386	101.44
0% query, 99% map	32	85.39	13.568	98.958	-	**98.958**
99% query, 0% map	65	112.96	27.56	140.52	-	140.52
	HOG					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
99% query, 99% map	70	65.28	0.301	65.581	0.0819	**65.662**
0% query, 99% map	83	72.5	0.3569	72.856	-	72.856
99% query, 0% map	90	83.93	0.387	84.317	-	84.317

Table 6.6: Total VPR time, in scenarios where the 17 places dataset is both uniformly and non-uniformly JPEG compressed.

	NetVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
90% query, 90% map	6	4.62	4.62	9.24	0.00752	**9.247**
0% query, 90% map	9	5.82	6.93	12.75	-	12.75
90% query, 0% map	13	7.77	10.01	17.78	-	17.78
	HybridNet					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
95% query, 95% map	8	25.46	2.88	28.34	0.00973	28.349
0% query, 95% map	8	25.46	2.88	28.34	-	**28.34**
95% query, 0% map	11	33.24	3.96	37.2	-	37.2
	RegionVLAD					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
95% query, 95% map	6	87.65	2.544	90.194	0.0073	**90.201**
0% query, 95% map	7	110.36	2.968	113.328	-	113.328
95% query, 0% map	9	150.63	3.816	154.446	-	154.446
	HOG					
JPEG Compression	K	t_m	t'_e	t_{VPR}	t'_c	t_{total}
80% query, 80% map	26	233.34	0.1118	233.4518	0.0414	**233.493**
0% query, 80% map	61	458.2	0.2623	458.462	-	458.462
80% query, 0% map	54	423.28	0.2322	423.512	-	423.512

6.5 Summary

To compensate for the reduction in VPR performance resulted from introducing high levels of JPEG compression, this chapter introduces sequence-based filtering in several well-established single-frame-based VPR techniques. The sequence length that results in a perfect place matching performance is reported for every descriptor throughout the entire spectrum of JPEG compression. To facilitate decentralised VPR applications where the limited bandwidth can impede the VPR process, the amount of data required to be transferred by each descriptor is analysed. Moreover, an investigation of the time required to perform VPR is provided. Our results show that a JPEG compressed image is often smaller in size when compared with an image descriptor and should be transmitted instead, in scenarios where limited bandwidth is available for VPR. Our experiments also conclude that it is often advantageous to compress the query images to the same quality of the map, leading to a more efficient VPR performance in changing environments.

Chapter 7

Data-Efficient VPR Using Low Resolution Images

Images incorporate a wealth of information from a robot's surroundings. With the widespread availability of compact cameras, visual information has become increasingly popular for addressing the localisation problem. While many applications use high-resolution cameras and high-end systems to achieve optimal place-matching performance, low-end commercial systems face limitations due to resource constraints and relatively low-resolution and low-quality cameras. In comparison with chapter 5 and chapter 6 which study the effects of JPEG compression for VPR applications, this chapter analyses the effects of image resolution on the accuracy and robustness of well-established handcrafted VPR pipelines. Handcrafted designs have low computational demands and can flexibly adapt to different image resolutions, making them a suitable approach to scale to any image source and to operate under resource limitations. This chapter aims to help academic researchers and companies in the hardware and software industry co-design VPR solutions and expand the use of VPR algorithms in commercial products.

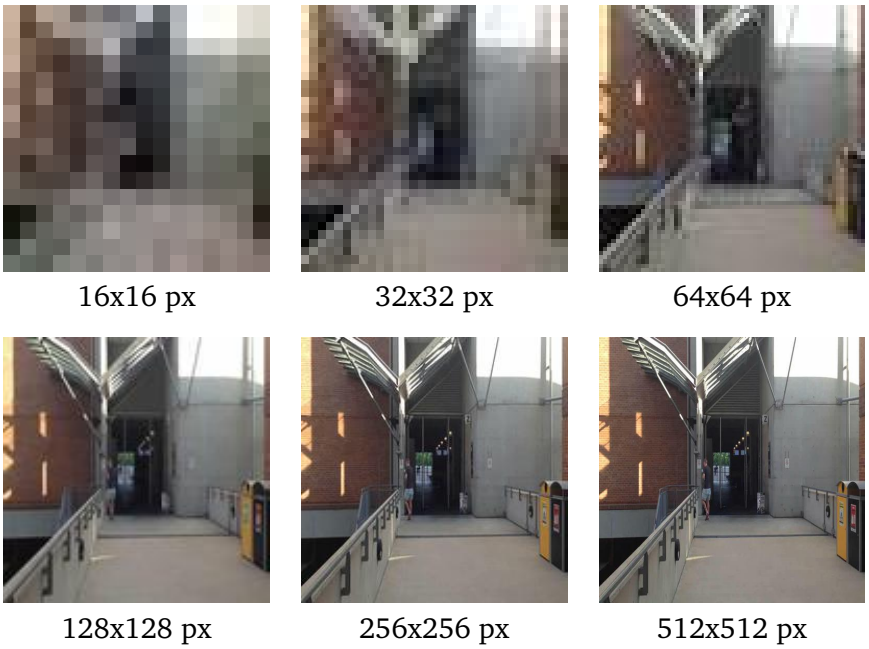


Figure 7.1: The same image resized to various resolutions.

7.1 Introduction

With the advances in technology made in the last decade, image and video capturing devices became exceptional in reproducing a higher quality representation of our surroundings. To achieve high place matching performance, VPR applications usually employ high-end systems and advanced cameras [8]. However, low-end commercial products are computationally limited and have low-resolution cameras. Thus, the deployment of robust but computationally demanding VPR methods is restricted on such platforms, as identified in [11, 33]. Hence, handcrafted VPR techniques are suitable to be deployed on resource constrained platforms, due to their computationally efficient nature. In addition to their low computational requirements, handcrafted VPR techniques can adapt to various image resolutions, which makes them attractive for VPR applications on resource-constrained platforms with low-resolution cameras. Moreover, as a lower-resolution image is visually different from its high-resolution version (refer to Fig. 7.1), this chapter analyses the optimal image resolution for different handcrafted descriptors. The focus of this work is mainly towards global feature descriptors, as local feature descriptors are unable to detect keypoints in small images. Therefore, they are not suitable to operate on small resolution images, as later shown in sub-section 7.3.1. The aim of this chapter is to reduce the image resolution to facilitate VPR applications on resource-constrained commercial platforms. In summary, our contributions are as follows:

Table 7.1: The size of each dataset in Megabytes (MB) resized to various resolutions.

	Image Resolution [px]						
Dataset	**16x16**	**32x32**	**64x64**	**128x128**	**256x256**	**512x512**	**1024x1024**
17 places	0.671	0.872	1.5	3.3	8.4	21.8	57.9
Campus Loop	0.151	0.194	0.339	0.845	2.7	9.3	28.7
Gardens Point	0.442	0.573	1	2.5	7.5	23.1	63.9
Nordland	0.25	0.311	0.512	1.2	3.4	10.1	29.8
SYNTHIA	0.296	0.379	0.647	1.5	4.6	16.2	56.9

- An assessment of the place matching performance of several well-established handcrafted VPR techniques on various image resolutions. We employ several datasets to enable a VPR performance comparison in real-world scenarios, under illumination, viewpoint and seasonal variations.
- We report the total time required to perform VPR for each descriptor, showing how a reduced image resolution results in a more efficient VPR process. We also perform a trade-off analysis between performance and computation, showing the best descriptor that should be selected depending on the image resolution.

7.2 Experimental Setup

7.2.1 VPR Time

For low-end commercial products which are computationally limited, it is important to determine the optimal technique in terms of resource utilisation. Hence, in this work we utilise the total time required to perform VPR (t_{VPR}) as a measurement of computational efficiency. The t_{VPR} of each technique is determined by adding the encoding time t_e with the matching time t_m as follows:

$$t_{VPR} = t_e + t_m \tag{7.1}$$

7.2.2 Performance Metric

To evaluate the VPR performance on various image resolutions, the percentage of correctly matched images is utilised as previously discussed in sub-section 2.7.2.

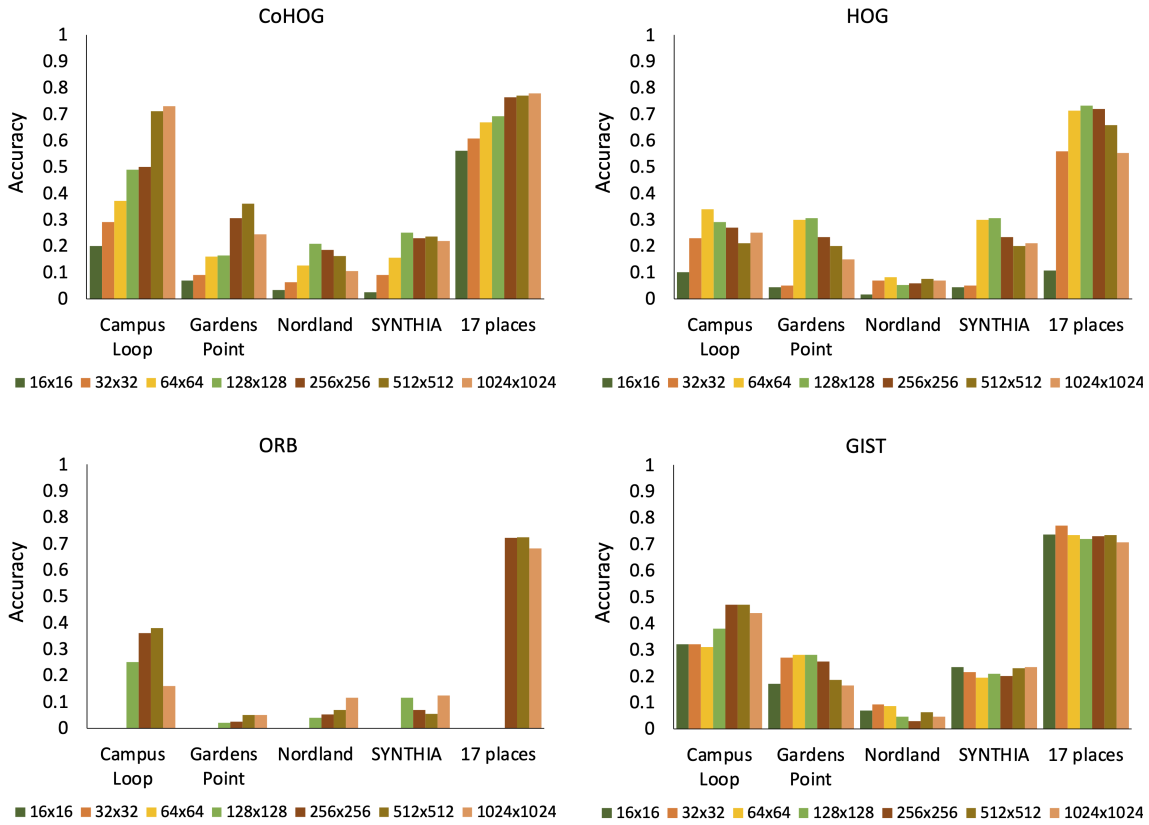


Figure 7.2: The accuracy of all VPR techniques on each resized dataset.

7.2.3 VPR Techniques

A selection of four well-established VPR techniques have been employed in this work including: HOG [65], CoHOG [67], ORB [53] and GIST [60]. For HOG, a cell and block size of 16x16 pixels was utilised, with a total of 9 histogram bins [7]. The remaining VPR techniques have been utilised as presented by their authors, with no additional changes being made to neither technique.

7.2.4 Test Datasets

In this chapter, five well-established VPR datasets (presented in sub-section 2.7.1) are employed to present our findings. These are as follows: Campus Loop [81]; Gardens Point [39], utilising *day_left* as query and *night_right* as reference images; Nordland [78]; SYNTHIA [128] and 17 places [129].

To enable a place matching performance comparison of each technique employed, the above mentioned datasets have been resized to several image resolutions (values presented

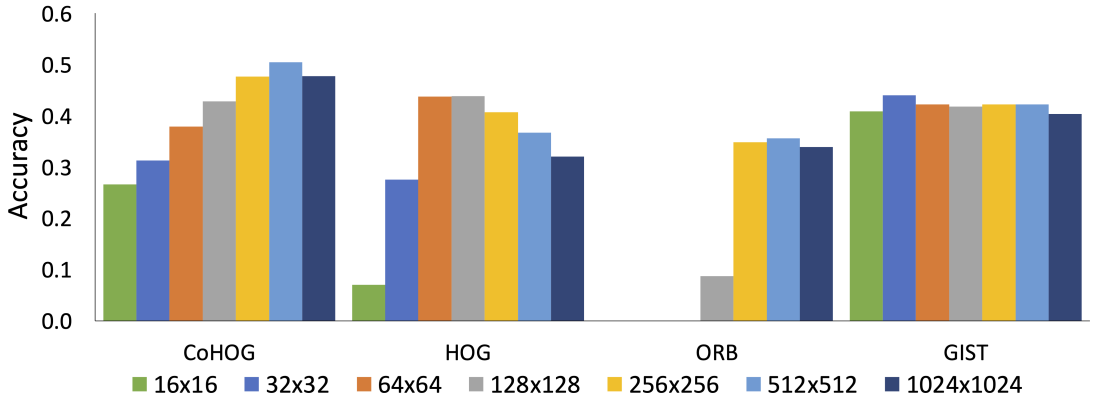
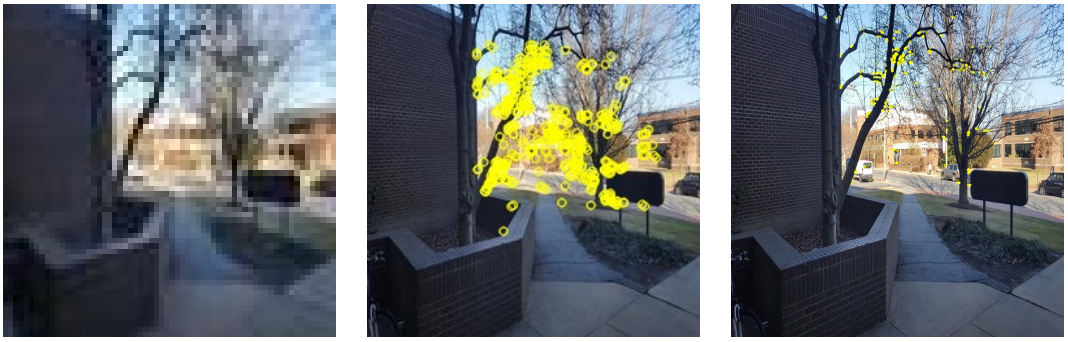


Figure 7.3: The average accuracy of each technique on the combined datasets.



64x64 px 256x256 px 1024x1024 px

Figure 7.4: Keypoints found in the same image at several distinct resolutions, as determined by ORB descriptor.

in pixels (px)), ranging from 16x16 px to 1024x1024 px. To resize the images to different resolutions, the Python Imaging Library (PIL[1]) has been utilised. The resizing process is presented in Algorithm 6. The values for new_width and new_height are specified by the user and represent the width and height values for the resized image. The resizing process is performed by the $resize$[2] function. Fig. 7.1 presents some sample images taken from the Gardens Point *day_left* dataset resized to various image resolutions. Table 7.1 presents the size in Megabytes of each resized dataset.

[1]https://pypi.org/project/Pillow/
[2]https://pillow.readthedocs.io/en/latest/_modules/PIL/Image.html#Image.resize

Algorithm 6: The algorithm utilised for image resizing is presented here.

new_width = $User\ Defined\ Constant$

new_height = $User\ Defined\ Constant$

```
// Resize the original image (img)
```

resized_img = img.resize((new_width, new_height))

Save resized_img

Table 7.2: The encoding time in milliseconds (ms) of a query image, for each VPR technique.

VPR Technique	**Image Resolution [px]**						
	16x16	**32x32**	**64x64**	**128x128**	**256x256**	**512x512**	**1024x1024**
CoHOG	14.5	15.3	16.6	19	30.6	77.3	260.1
HOG	0.104	0.236	1.307	0.514	1.585	6.578	31.32
ORB	-	-	-	0.86	2.49	6.171	17.5
GIST	0.967	2	9.807	27.561	153.99	708.02	4618.1

7.3 Results and Analysis

7.3.1 Place Matching Performance

The performance of all VPR techniques on every resized dataset is presented in Fig. 7.2. In contrast with the VPR accuracy of HOG and GIST which peaks towards smaller images, CoHOG benefits from an increased image resolution. Moreover, as CoHOG is designed to handle lateral shifts in camera movement, this technique achieves high accuracy on 17 places and Campus Loop datasets, while utilising a higher image resolution than the rest of the techniques (1024x1024 px). This trend is also emphasized in Fig. 7.3, which presents the average performance for each technique on all presented datasets, where the accuracy for each image resolution is weighted with regards to the number of images in the dataset. CoHOG achieves the highest place matching performance on datasets resized to 512x512 px. For GIST, the highest accuracy is reported on the datasets resized to 32x32 px. HOG achieves similar levels of performance on both 64x64 px and 128x128 px resized datasets, as seen in Fig. 7.3. It is important to note that ORB cannot work with small image resolutions, as previously mentioned in section 7.1. This happens because no keypoints are detected in images, or the image is smaller than the descriptor patch. In our experiments, ORB cannot work with image resolutions of less than 128x128 px. Moreover, for 17 places dataset, ORB does not find any

Table 7.3: The matching time in milliseconds (ms), for each VPR technique.

CoHOG	**16x16**	**32x32**	**64x64**	**128x128**	**256x256**	**512x512**	**1024x1024**
Campus Loop	0.93	0.99	1.35	4.13	28.7	299	5403.1
Gardens Point	1.87	1.96	2.83	8.25	56.5	586	10714.7
Nordland	1.69	1.75	2.25	7.09	48.8	514	9171.97
SYNTHIA	2.03	2.15	2.61	8.35	56.7	601	10735.55
17 places	4.47	4.79	5.55	18.91	128.96	1352.07	24478.34
HOG	**16x16**	**32x32**	**64x64**	**128x128**	**256x256**	**512x512**	**1024x1024**
Campus Loop	1	1.04	1.08	1.33	2.01	5.46	18.2
Gardens Point	2.13	2.34	2.38	2.75	4.56	11.2	35.85
Nordland	1.7	1.83	2.09	2.68	4.23	9.19	31.33
SYNTHIA	2.11	2.45	2.56	2.94	4.39	11.4	35.45
17 places	4.5	4.56	4.71	5.56	9.83	24.93	80.16
ORB	**16x16**	**32x32**	**64x64**	**128x128**	**256x256**	**512x512**	**1024x1024**
Campus Loop	-	-	-	21.6	110.7	145	151
Gardens Point	-	-	-	43.05	223.8	300.95	285.7
Nordland	-	-	-	19.47	137.96	239.01	238.54
SYNTHIA	-	-	-	30.75	197.7	294.25	285.5
17 places	-	-	-	-	396.17	618.8	607.51
GIST	**16x16**	**32x32**	**64x64**	**128x128**	**256x256**	**512x512**	**1024x1024**
Campus Loop	0.78	0.83	0.978	1.06	0.885	0.92	0.88
Gardens Point	1.67	1.675	1.625	1.57	1.725	1.65	1.62
Nordland	1.715	1.529	1.616	1.389	1.389	1.372	1.366
SYNTHIA	1.6	1.525	1.67	1.805	1.525	1.505	1.625
17 places	3.54	3.45	3.67	3.53	3.83	3.95	4.56

keypoints in image resolutions of less than 256x256 px. Fig. 7.4 shows the keypoint locations of ORB at different image resolutions. It can be seen that by reducing the image resolution to 64x64 px, ORB fails to detect any of the previously identified keypoints in the presented environment.

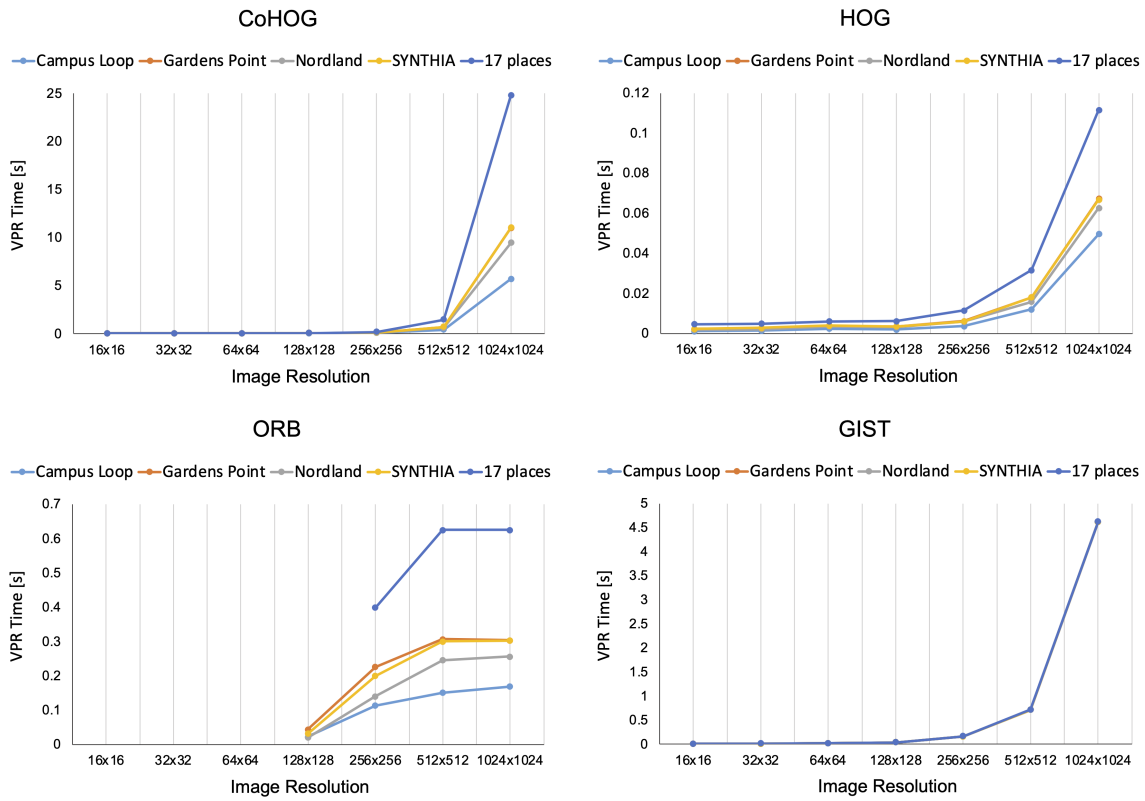


Figure 7.5: The VPR time (refer to equation (7.1)) in seconds (s) of all VPR techniques on various image resolutions.

7.3.2 Analysis on the Time Required to Perform VPR

This sub-section performs an analysis on the total time required to perform VPR. Table 7.2 presents the encoding time t_e and Table 7.3 presents the matching time t_m of all VPR techniques. Moreover, the VPR time (refer to equation (7.1)) of every technique is presented in Fig. 7.5. We have previously discussed in sub-section 7.3.1 that CoHOG achieves increased levels of place matching performance utilising a higher image resolution. However, as CoHOG presents a longer matching time t_m when utilising a higher image resolution, its VPR time is drastically increased, as observed in Fig. 7.5 on the 17 places dataset. When utilising an image resolution of 128x128 px and above, GIST achieves high encoding times when compared to the remaining VPR techniques, as reported in Table 7.2. In contrast with HOG, ORB and CoHOG where the descriptor size changes depending on the image resolution, the descriptor size of GIST always remains constant, therefore its matching time (refer to Table 7.3) remains similar for every image resolution. Thus, the t_{VPR} of GIST does not differ significantly from one dataset to another, as shown in Fig. 7.5. Hence, GIST should be selected for VPR applica-

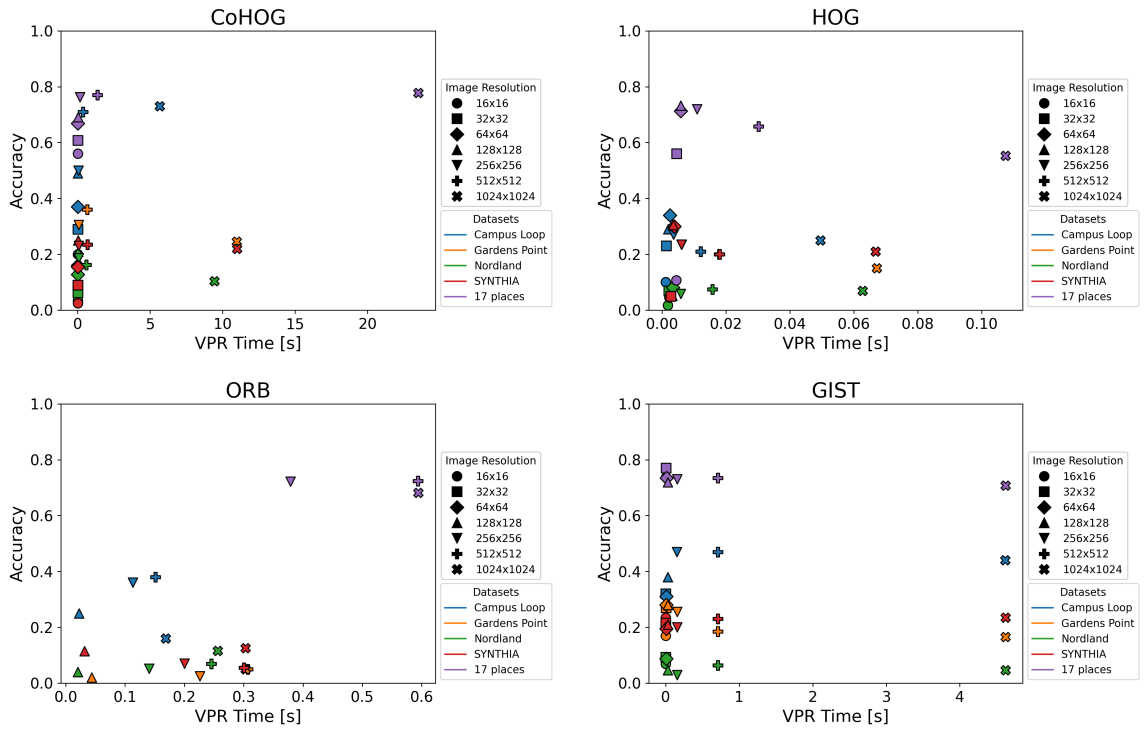


Figure 7.6: The accuracy of each technique and the corresponding VPR time for each resized dataset.

tions with a focus on fast processing times. However, if the aim is towards VPR performance, CoHOG should be utilised instead.

7.3.3 Performance and Computation Trade-off Analysis

As utilising a lower image resolution generally results in a decrease in t_{VPR} (refer to Fig. 7.5), this section performs a trade-off analysis between VPR performance and time. Fig. 7.6 presents the VPR time (on x-axis) required to perform place matching on each resized dataset and the corresponding accuracy (on y-axis) for each VPR technique. As previously mentioned in sub-section 7.3.1, CoHOG generally achieves higher place matching performance while using larger image resolutions, albeit at a considerable increase in t_{VPR}. Thus, in comparison with VPR techniques such as HOG and GIST which perform better whilst utilising a lower image resolution, the accuracy of CoHOG is noticeably higher on datasets with high resolution images, albeit at a considerable increase in t_{VPR}, as shown in Fig. 7.6.

7.4 Summary

This chapter presents an in-depth study on the effects of image resolution on the place matching performance of several well-established handcrafted VPR techniques. We confirmed that local feature descriptors are unable to operate on very small images, hence the focus of this work is mostly related to global feature descriptors. We utilise several VPR datasets to present our results, and show that the time required to perform VPR is reduced with a decrease in image resolution. Moreover, this chapter performs a trade-off analysis between performance and computation, showing how utilising a lower image resolution results in a more efficient VPR process to allow efficient deployment on low-end commercial products.

Chapter 8

Concluding Remarks and Future Directions

Despite the significant progress in the development of robust and accurate algorithms during the past decades, Visual Place Recognition (VPR) still remains a challenging task. The ability to handle dynamic large-scale environments and the presence of significant illumination, viewpoint and seasonal variations in the environment can drastically decrease the performance of a VPR descriptor. However, the continued advancement in computing power, together with the ever-growing availability of large-scale VPR datasets will allow new methods to emerge, in an effort to overcome these challenging problems. Moreover, as VPR is a constantly improving field, it is expected that these improvements will enable a considerably more accurate and reliable place matching performance in changing environments in the years to come.

This chapter presents the concluding remarks, together with the contributions of this thesis, our findings and extensions of the proposed work. More specifically, section 8.1 presents the novel research conducted and major contributions brought in this thesis. Section 8.2 presents future avenues of research for both sequence-based filtering and JPEG compression in the field of Visual Place Recognition. We hope that this thesis has established the foundation of sequence-based filtering and JPEG compression within VPR, whilst providing a deeper understanding on how to develop better VPR pipelines. We also hope that further research can build upon our findings and bring further contributions to this field.

8.1 Thesis Contributions

In an attempt to overcome some of the challenges in the field of Visual Place Recognition in changing environments, this thesis proposes several contributions that are provided below.

In chapter 3, to combat the extreme computational demands of CNN-based VPR techniques and the limitations of utilising a constant sequence length in sequence-based filtering contexts, we propose a sequence-based, training-less VPR technique for changing environments. The proposed technique, entitled ConvSequential-SLAM, is successfully able to adapt its sequence length depending on the environment, while at the same time achieving state-of-the-art place matching performance on datasets that contain viewpoint and appearance variations. Moreover, we developed a matching schema that is agnostic to the underlying single-frame-based VPR technique and can enable an efficient comparison of the effects of sequence-based filtering on top of single-frame-based VPR techniques.

Chapter 4 presents a systematic study of sequence-based filtering for visual route-based navigation. More specifically, we employ the matching schema proposed in ConvSequential-SLAM to analyse the benefits and trade-offs of sequence-based filtering. We show how light-weight techniques can replace more complex VPR descriptors to perform VPR more efficiently. We hope that the insights presented in this chapter will enable a better understanding of the advantages and limitations of deploying sequence-based filtering on single-frame-based VPR techniques, for designing better VPR systems.

In chapter 5, we shift our attention from sequence-based filtering to an area which has been previously overlooked by the research community, *viz.* the study of JPEG compression for VPR applications. JPEG is designed to reduce the clarity and size of an image, whilst having a minimal impact on the human perception system. However, we show that it can also be employed in decentralised VPR applications to reduce the amount of data transferred over a communication channel. While VPR performance is reduced especially in the higher spectrum of compression, we observed that JPEG affects more those methods which are designed to deal with viewpoint changes rather than appearance changes. To overcome the drastic reduction in VPR performance when utilising extremely JPEG compressed images, we demonstrate how a fine-tuned CNN-based VPR technique is able to achieve more consistent place matching performance.

In chapter 6, we utilise the sequence-based filtering schema proposed in ConvSequential-SLAM to overcome the performance degradation resulted from utilising high ratios of JPEG compression. We show that the amount of data transferred over a communication channel is

extremely reduced towards the high spectrum of JPEG compression, even at the expense of an increased sequence length. Moreover, when comparing the size of a compressed image with the size of the image descriptor, we found that the former is often smaller, hence should be transferred instead. When analysing the amount of time required to perform VPR whilst utilising a sequence-based filtering approach on highly JPEG compressed data, our experiments determined that more efficient VPR can be executed when the query and reference images are compressed to the same ratio.

In chapter 7, we analyse the effects of image resolution on the performance and robustness of several well-established handcrafted VPR techniques. Our analysis confirms that local feature descriptors are not suitable to operate on low-resolution images, as they do not detect any keypoints in small images. Therefore, the main focus of this chapter is towards global feature descriptors. The total time required to perform VPR is analysed, and we show that it is correlated with the image resolution. A trade-off analysis between performance and computation is performed for every handcrafted VPR technique, to enable efficient deployment of VPR solutions on low-end commercial products.

8.2 Future Directions

While several contributions have been proposed in this thesis to address some of the already existing challenges in VPR, extensive work is still required to achieve robust robot perception in its deepest sense. However, several interesting research directions emerged following this thesis. These are as provided below.

Although ConvSequential-SLAM utilises HOG to compute the image descriptors, more robust underlying image similarity algorithms can be employed. Another possible future direction for improving this work is to cater for dynamic objects and confusing features coming from trees and vegetation in outdoor environments.

The simple matching schema utilised in chapter 4 was employed to highlight the benefits of utilising sequences of images for VPR. A natural extension of this work is comparing different matching schema. While we demonstrated that VPR accuracy generally benefits from using a sequence of images to find a place, sequence matching has some more strict requirements than single-matching approaches. The most relevant requirement is in regards to the velocity of the traverses. If the velocity of the reference sequence is too different from that of the query, the matching might fail [72]. Thus, the analysis proposed in this chapter could be

extended to more complex sequence-based matching techniques to understand whether the trade-off between the sequence length, VPR performance and computational cost are affected by the matching method.

We have shown in chapter 5 that a fine-tuned CNN on highly JPEG compressed data can achieve more consistent place matching performance than non-optimised techniques. Hence, this research can be further extended by analysing the VPR performance of other fine-tuned CNNs on highly JPEG compressed data, to determine the best performing descriptor in different scenarios. While our experiments concluded that JPEG compression affects more those descriptors that are designed to handle viewpoint changes rather than appearance changes, additional research needs to be conducted to accurately determine the exact reason. Moreover, the analysis conducted in chapter 6 can be utilised as the basis for an adaptive sequence-based VPR system capable of switching between different VPR techniques and sequence lengths, for achieving the best possible VPR performance in regards to the available bandwidth for VPR. The performance of ConvSequential-SLAM has not been investigated on JPEG compression data. Hence, it would be interesting to determine whether this technique achieves improved performance over the methods tested in chapter 6. Future work can also explore new methods to optimise VPR techniques for JPEG imagery. These can include re-training or re-calibrating VPR techniques specifically for handling highly JPEG compressed data. Moreover, it would be interesting to investigate a sequence-based video compression codec such as H.265 [148], as an alternative to sequence-based VPR. We strongly believe that a VPR-specific image compression line of research would greatly benefit VPR applications, and we hope that the limitations highlighted in these studies on JPEG compression can therefore be overcome.

Apart from the computational benefits of utilising low-resolution images presented in chapter 7, this research also has potential benefits for visual privacy. Thus, VPR could potentially be performed on images of sufficiently low-resolution such as they do not compromise visual privacy. An extension of the work proposed in chapter 7 can investigate VPR using low-resolution images in environments where delicate visual information is present, such as faces in crowded environments and car plate numbers.

Bibliography

[1] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. M. Paixao, F. Mutz, *et al.*, "Self-driving cars: A survey," *Expert Systems with Applications*, vol. 165, p. 113816, 2021.

[2] M. Buehler, K. Iagnemma, and S. Singh, *The 2005 DARPA grand challenge: the great robot race*, vol. 36. springer, 2007.

[3] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, *et al.*, "Stanley: The robot that won the darpa grand challenge," *Journal of field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[4] M. Buehler, K. Iagnemma, and S. Singh, *The DARPA urban challenge: autonomous vehicles in city traffic*, vol. 56. springer, 2009.

[5] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer, *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[6] M. Zaffar, S. Ehsan, M. Milford, and K. D. McDonald-Maier, "Memorable maps: A framework for re-defining places in visual place recognition," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[7] M. Zaffar, S. Garg, M. Milford, J. Kooij, D. Flynn, K. McDonald-Maier, and S. Ehsan, "Vpr-bench: An open-source visual place recognition evaluation framework with quantifiable viewpoint and appearance change," *International Journal of Computer Vision*, vol. 129, no. 7, pp. 2136–2174, 2021.

[8] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2016.

[9] Z. Chen, O. Lam, A. Jacobson, and M. Milford, "Convolutional neural network-based place recognition," *arXiv preprint arXiv:1411.1509*, 2014.

[10] M. Zaffar, A. Khaliq, S. Ehsan, M. Milford, and K. McDonald-Maier, "Levelling the playing field: A comprehensive comparison of visual place recognition approaches under changing conditions," *arXiv preprint arXiv:1903.09107*, 2019.

[11] M. Zaffar, A. Khaliq, S. Ehsan, M. Milford, K. Alexis, and K. McDonald-Maier, "Are state-of-the-art visual place recognition techniques any good for aerial robotics?," *arXiv preprint arXiv:1904.07967*, 2019.

[12] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[13] Z. Chen, A. Jacobson, N. Sünderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, "Deep learning features at scale for visual place recognition," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3223–3230, IEEE, 2017.

[14] A. Torii, J. Sivic, T. Pajdla, and M. Okutomi, "Visual place recognition with repetitive structures," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 883–890, 2013.

[15] F. Maffra, L. Teixeira, Z. Chen, and M. Chli, "Real-time wide-baseline place recognition using depth completion," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1525–1532, 2019.

[16] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[17] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "The revisiting problem in simultaneous localization and mapping: A survey on visual loop closure detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 19929–19953, 2022.

[18] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[19] G. H. Lee, F. Fraundorfer, and M. Pollefeys, "Robust pose-graph loop-closures with expectation-maximization," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 556–563, IEEE, 2013.

[20] L. Xie, S. Wang, A. Markham, and N. Trigoni, "Graphtinker: Outlier rejection and inlier injection for pose graph slam," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6777–6784, IEEE, 2017.

[21] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: an open-source library for real-time metric-semantic localization and mapping," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1689–1696, IEEE, 2020.

[22] M. Xu, T. Fischer, N. Sünderhauf, and M. Milford, "Probabilistic appearance-invariant topometric localization with new place awareness," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6985–6992, 2021.

[23] A. Angeli, S. Doncieux, J.-A. Meyer, and D. Filliat, "Real-time visual loop-closure detection," in *2008 IEEE international conference on robotics and automation*, pp. 1842–1847, IEEE, 2008.

[24] G. Schindler, M. Brown, and R. Szeliski, "City-scale location recognition," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, IEEE, 2007.

[25] J. Röwekämper, C. Sprunk, G. D. Tipaldi, C. Stachniss, P. Pfaff, and W. Burgard, "On the position accuracy of mobile robot localization based on particle filters combined with scan matching," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3158–3164, IEEE, 2012.

[26] I. Ulrich and I. Nourbakhsh, "Appearance-based place recognition for topological localization," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 2, pp. 1023–1029, Ieee, 2000.

[27] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[28] C. Cadena, D. Gálvez-López, J. D. Tardós, and J. Neira, "Robust place recognition with stereo sequences," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 871–885, 2012.

[29] B. Kuipers, "Modeling spatial knowledge," *Cognitive science*, vol. 2, no. 2, pp. 129–153, 1978.

[30] Z. Chen, A. Jacobson, U. M. Erdem, M. E. Hasselmo, and M. Milford, "Multi-scale bio-inspired place recognition," in *2014 IEEE international conference on robotics and automation (ICRA)*, pp. 1895–1901, IEEE, 2014.

[31] Z. Chen, S. Lowry, A. Jacobson, M. E. Hasselmo, and M. Milford, "Bio-inspired homogeneous multi-scale place recognition," *Neural Networks*, vol. 72, pp. 48–61, 2015.

[32] I. Kostavelis, K. Charalampous, A. Gasteratos, and J. K. Tsotsos, "Robot navigation via spatial and temporal coherent semantic maps," *Engineering Applications of Artificial Intelligence*, vol. 48, pp. 173–187, 2016.

[33] B. Ferrarini, M. Waheed, S. Waheed, S. Ehsan, M. Milford, and K. D. McDonald-Maier, "Visual place recognition for aerial robotics: Exploring accuracy-computation trade-off for local image descriptors," in *2019 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, pp. 103–108, 2019.

[34] M.-A. Tomită, M. Zaffar, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Convsequential-slam: A sequence-based, training-less visual place recognition technique for changing environments," *IEEE Access*, vol. 9, pp. 118673–118683, 2021.

[35] M.-A. Tomită, M. Zaffar, B. Ferrarini, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Sequence-based filtering for visual route-based navigation: Analyzing the benefits, trade-offs and design choices," *IEEE Access*, vol. 10, pp. 81974–81987, 2022.

[36] M.-A. Tomita, B. Ferrarini, M. Milford, K. McDonald-Maier, and S. Ehsan, "Visual place recognition with low-resolution images," *arXiv preprint arXiv:2305.05776*, 2023.

[37] M.-A. Tomita, B. Ferrarini, M. Milford, K. McDonald-Maier, and S. Ehsan, "Data efficient visual place recognition using extremely jpeg-compressed images," *arXiv preprint arXiv:2209.08343*, 2022.

[38] M.-A. Tomita, B. Ferrarini, M. Milford, K. McDonald-Maier, and S. Ehsan, "Data-efficient sequence-based visual place recognition with highly compressed jpeg images," *arXiv preprint arXiv:2302.13314*, 2023.

[39] N. Sünderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance

of convnet features for place recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4297–4304, IEEE, 2015.

[40] Z. Zeng, J. Zhang, X. Wang, Y. Chen, and C. Zhu, "Place recognition: An overview of vision perspective," *Applied Sciences*, vol. 8, no. 11, p. 2257, 2018.

[41] C. Masone and B. Caputo, "A survey on deep visual place recognition," *IEEE Access*, vol. 9, pp. 19516–19547, 2021.

[42] S. Garg, T. Fischer, and M. Milford, "Where is your place, visual place recognition?," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21* (Z.-H. Zhou, ed.), pp. 4416–4425, International Joint Conferences on Artificial Intelligence Organization, 8 2021. Survey Track.

[43] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1150–1157 vol.2, 1999.

[44] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features.," in *Computer Vision and Image Understanding - CVIU*, vol. 110, pp. 404–417, 01 2006.

[45] S. Se, D. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *The international Journal of robotics Research*, vol. 21, no. 8, pp. 735–758, 2002.

[46] H. Andreasson and T. Duckett, "Topological localization for mobile robots using omnidirectional vision and local features," *IFAC Proceedings Volumes*, vol. 37, no. 8, pp. 36–41, 2004.

[47] E. Stumm, C. Mei, and S. Lacroix, "Probabilistic place recognition with covisibility maps," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4158–4163, IEEE, 2013.

[48] J. Košecká, F. Li, and X. Yang, "Global localization and relative positioning based on scale-invariant keypoints," *Robotics and Autonomous Systems*, vol. 52, no. 1, pp. 27–38, 2005.

[49] A. C. Murillo, J. J. Guerrero, and C. Sagues, "Surf features for efficient robot localization with omnidirectional images," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 3901–3907, IEEE, 2007.

[50] M. Cummins and P. Newman, "Appearance-only slam at large scale with fab-map 2.0," *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1100–1123, 2011.

[51] W. Maddern, M. Milford, and G. Wyeth, "Cat-slam: probabilistic localisation and mapping using a continuous appearance-based trajectory," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 429–451, 2012.

[52] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, "Brief: Computing a local binary descriptor very fast," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1281–1298, 2011.

[53] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International conference on computer vision*, pp. 2564–2571, Ieee, 2011.

[54] M. Agrawal, K. Konolige, and M. R. Blas, "Censure: Center surround extremas for realtime feature detection and matching," in *European Conference on Computer Vision*, pp. 102–115, Springer, 2008.

[55] K. Konolige and M. Agrawal, "Frameslam: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066–1077, 2008.

[56] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

[57] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3304–3311, IEEE, 2010.

[58] A. Angeli, S. Doncieux, J.-A. Meyer, and D. Filliat, "Incremental vision-based topological slam," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1031–1036, Ieee, 2008.

[59] A. Oliva and A. Torralba, "Chapter 2 building the gist of a scene: the role of global image features in recognition," in *Visual Perception* (S. Martinez-Conde, S. Macknik, L. Martinez, J.-M. Alonso, and P. Tse, eds.), vol. 155 of *Progress in Brain Research*, pp. 23 – 36, Elsevier, 2006.

[60] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[61] A. C. Murillo and J. Kosecka, "Experiments in place recognition using gist panoramas," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pp. 2196–2203, IEEE, 2009.

[62] G. Singh and J. Kosecka, "Visual loop closing using gist descriptors in manhattan world," in *ICRA Omnidirectional Vision Workshop*, pp. 4042–4047, 2010.

[63] C. Siagian and L. Itti, "Biologically inspired mobile robot vision localization," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 861–873, 2009.

[64] H. Badino, D. Huber, and T. Kanade, "Real-time topometric localization," in *2012 IEEE International Conference on Robotics and Automation*, pp. 1635–1642, IEEE, 2012.

[65] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886–893, IEEE, 2005.

[66] C. McManus, B. Upcroft, and P. Newmann, "Scene signatures: Localised and point-less features for localisation," *Proc. Robot.: Sci. Syst*, 2014.

[67] M. Zaffar, S. Ehsan, M. Milford, and K. McDonald-Maier, "Cohog: A light-weight, compute-efficient, and training-free visual place recognition technique for changing environments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1835–1842, 2020.

[68] B. Ferrarini, M. Waheed, S. Waheed, S. Ehsan, M. J. Milford, and K. D. McDonald-Maier, "Exploring performance bounds of visual place recognition using extended precision," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1688–1695, 2020.

[69] M. Waheed, M. Milford, K. McDonald-Maier, and S. Ehsan, "Improving visual place recognition performance by maximising complementarity," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5976–5983, 2021.

[70] M. Waheed, M. Milford, K. McDonald-Maier, and S. Ehsan, "Switchhit: A probabilistic, complementarity-based switching system for improved visual place recogni-

tion in changing environments," in *Accepted to 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022. preprint on webpage at https://arxiv.org/abs/2203.00591.

[71] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[72] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *2012 IEEE International Conference on Robotics and Automation*, pp. 1643–1649, IEEE, 2012.

[73] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[74] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5297–5307, 2016.

[75] N. Sünderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford, "Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free," *Robotics: Science and Systems XI:*, pp. 1–10, 2015.

[76] A. Glover, "Day and night, left and right," *Zenodo DOI*, vol. 10, 2014.

[77] P. Neubert, N. Sünderhauf, and P. Protzel, "Appearance change prediction for long-term navigation across seasons," in *2013 European Conference on Mobile Robots*, pp. 198–203, IEEE, 2013.

[78] S. Skrede, "Nordland dataset," 2013. Available online at: *https://nrkbeta.no/2013/01/15/nordlandsbanen-minute-by-minute-season-by-season/*.

[79] Z. Chen, F. Maffra, I. Sa, and M. Chli, "Only look once, mining distinctive landmarks from convnet for visual place recognition," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9–16, IEEE, 2017.

[80] A. Khaliq, S. Ehsan, M. Milford, and K. McDonald-Maier, "Camal: Context-aware multi-scale attention framework for lightweight visual place recognition," *arXiv preprint arXiv:1909.08153*, 2019.

[81] N. Merrill and G. Huang, "Lightweight unsupervised deep loop closure," *arXiv preprint arXiv:1805.07703*, 2018.

[82] A. Torii, R. Arandjelovic, J. Sivic, M. Okutomi, and T. Pajdla, "24/7 place recognition by view synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1808–1817, 2015.

[83] A. Khaliq, S. Ehsan, Z. Chen, M. Milford, and K. McDonald-Maier, "A holistic visual place recognition approach using lightweight cnns for significant viewpoint and appearance changes," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 561–569, 2019.

[84] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[85] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superglue: Learning feature matching with graph neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4938–4947, 2020.

[86] M. A. Uy and G. H. Lee, "Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4470–4479, 2018.

[87] S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer, "Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14141–14152, 2021.

[88] B. Cao, A. Araujo, and J. Sim, "Unifying deep local and global features for image search," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pp. 726–743, Springer, 2020.

[89] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12716–12725, 2019.

[90] B. Ferrarini, M. J. Milford, K. D. McDonald-Maier, and S. Ehsan, "Binary neural networks for memory-efficient and effective visual place recognition in changing environments," *IEEE Transactions on Robotics*, pp. 1–15, 2022.

[91] B. Ferrarini, M. Milford, K. D. McDonald-Maier, and S. Ehsan, "Highly-efficient binary neural networks for visual place recognition," in *Accepted to 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022. preprint on webpage at https://arxiv.org/abs/2202.12375.

[92] B. Arcanjo, B. Ferrarini, M. Milford, K. D. McDonald-Maier, and S. Ehsan, "An efficient and scalable collection of fly-inspired voting units for visual place recognition in changing environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2527–2534, 2022.

[93] K. Ho and P. Newman, "Detecting loop closure with scene sequences.," *International journal of computer vision*, vol. 74, no. 3, 2007.

[94] E. Pepperell, P. I. Corke, and M. J. Milford, "All-environment visual place recognition with smart," in *2014 IEEE international conference on robotics and automation (ICRA)*, pp. 1612–1618, IEEE, 2014.

[95] M. Yang, J. Mao, X. He, L. Zhang, and X. Hu, "A sequence-based visual place recognition method for aerial mobile robots," in *Journal of Physics: Conference Series*, vol. 1654, p. 012080, IOP Publishing, 2020.

[96] S. Garg and M. Milford, "Fast, compact and highly scalable visual place recognition through sequence-based matching of overloaded representations," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3341–3348, IEEE, 2020.

[97] E. Johns and G.-Z. Yang, "Feature co-occurrence maps: Appearance-based localisation throughout the day," in *2013 IEEE International Conference on Robotics and Automation*, pp. 3212–3218, IEEE, 2013.

[98] M. Chancán and M. Milford, "Deepseqslam: A trainable cnn+ rnn for joint global description and sequence-based place recognition," *arXiv preprint arXiv:2011.08518*, 2020.

[99] T. Naseer, L. Spinello, W. Burgard, and C. Stachniss, "Robust visual robot localization across seasons using network flows," in *Twenty-eighth AAAI conference on artificial intelligence*, 2014.

[100] O. Vysotska and C. Stachniss, "Relocalization under substantial appearance changes

using hashing," in *Proceedings of the IROS Workshop on Planning, Perception and Navigation for Intelligent Vehicles, Vancouver, BC, Canada*, vol. 24, 2017.

[101] P. Neubert, S. Schubert, and P. Protzel, "A neurologically inspired sequence processing model for mobile robot place recognition," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3200–3207, 2019.

[102] A.-D. Doan, Y. Latif, T.-J. Chin, Y. Liu, S.-F. Ch'ng, T.-T. Do, and I. Reid, "Visual localization under appearance change: filtering approaches," *Neural Computing and Applications*, vol. 33, no. 13, pp. 7325–7338, 2021.

[103] L. Bampis and A. Gasteratos, "Sequence-based visual place recognition: a scale-space approach for boundary detection," *Autonomous Robots*, vol. 45, no. 4, pp. 505–518, 2021.

[104] F. Xiong, Y. Ding, M. Yu, W. Zhao, N. Zheng, and P. Ren, "A lightweight sequence-based unsupervised loop closure detection," in *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2021.

[105] C. Fu, C. Xiang, C. Wang, and D. Cai, "Fast approximate nearest neighbor search with the navigating spreading-out graph," *arXiv preprint arXiv:1707.00143*, 2017.

[106] E. Stenborg, T. Sattler, and L. Hammarstrand, "Using image sequences for long-term visual localization," in *2020 International Conference on 3D Vision (3DV)*, pp. 938–948, 2020.

[107] X. Zhang, L. Wang, Y. Zhao, and Y. Su, "Graph-based place recognition in image sequences with cnn features," *Journal of Intelligent & Robotic Systems*, vol. 95, pp. 389–403, 2019.

[108] S. Garg, M. Vankadari, and M. Milford, "Seqmatchnet: Contrastive learning with sequence matching for place recognition & relocalization," in *Conference on Robot Learning*, pp. 429–443, PMLR, 2022.

[109] P. Hansen and B. Browning, "Visual place recognition using hmm sequence matching," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4549–4555, IEEE, 2014.

[110] F. Lu, B. Chen, X.-D. Zhou, and D. Song, "Sta-vpr: Spatio-temporal alignment for visual

place recognition," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4297–4304, 2021.

[111] S. Garg and M. Milford, "Seqnet: Learning descriptors for sequence-based hierarchical place recognition," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4305–4312, 2021.

[112] R. Mereu, G. Trivigno, G. Berton, C. Masone, and B. Caputo, "Learning sequential descriptors for sequence-based visual place recognition," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10383–10390, 2022.

[113] J. M. Facil, D. Olid, L. Montesano, and J. Civera, "Condition-invariant multi-view place recognition," *arXiv preprint arXiv:1902.09516*, 2019.

[114] F. Warburg, S. Hauberg, M. Lopez-Antequera, P. Gargallo, Y. Kuang, and J. Civera, "Mapillary street-level sequences: A dataset for lifelong place recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2626–2635, 2020.

[115] S. Garg, B. Harwood, G. Anand, and M. Milford, "Delta descriptors: Change-based place representation for robust visual localization," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5120–5127, 2020.

[116] J. Schoolcraft, A. Klesh, and T. Werne, "Marco: interplanetary mission development on a cubesat scale," *Space Operations: Contributions from the Global Community*, pp. 221–231, 2017.

[117] E. Nettleton, S. Thrun, H. Durrant-Whyte, and S. Sukkarieh, "Decentralised slam with low-bandwidth communication for teams of vehicles," in *Field and Service Robotics*, vol. 24, pp. 179–188, Springer, 2006.

[118] T. Cieslewski and D. Scaramuzza, "Efficient decentralized visual place recognition from full-image descriptors," in *2017 International symposium on multi-robot and multi-agent systems (MRS)*, pp. 78–82, IEEE, 2017.

[119] T. Cieslewski, S. Choudhary, and D. Scaramuzza, "Data-efficient decentralized visual slam," in *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 2466–2473, IEEE, 2018.

[120] A. Burguera and F. Bonin-Font, "An unsupervised neural network for loop detection in underwater visual slam," *Journal of Intelligent & Robotic Systems*, vol. 100, no. 3, pp. 1157–1177, 2020.

[121] P. Schmuck and M. Chli, "Multi-uav collaborative monocular slam," in *International Conference on Robotics and Automation (ICRA)*, pp. 3863–3870, IEEE, 2017.

[122] P. Schmuck, T. Ziegler, M. Karrer, J. Perraudin, and M. Chli, "Covins: Visual-inertial slam for centralized collaboration," in *2021 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 171–176, IEEE, 2021.

[123] T. Cieslewski and D. Scaramuzza, "Efficient decentralized visual place recognition using a distributed inverted index," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 640–647, 2017.

[124] A. Gautam and S. Mohan, "A review of research in multi-robot systems," in *2012 IEEE 7th International Conference on Industrial and Information Systems (ICIIS)*, pp. 1–5, 2012.

[125] V. Polizzi, R. Hewitt, J. Hidalgo-Carrió, J. Delaune, and D. Scaramuzza, "Data-efficient collaborative decentralized thermal-inertial odometry," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10681–10688, 2022.

[126] E. R. Boroson, R. Hewitt, N. Ayanian, and J.-P. de la Croix, "Inter-robot range measurements in pose graph optimization," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4806–4813, 2020.

[127] S. Van Der Walt, S. C. Colbert, and G. Varoquaux, "The numpy array: a structure for efficient numerical computation," *Computing in science & engineering*, vol. 13, no. 2, pp. 22–30, 2011.

[128] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[129] R. Sahdev and J. K. Tsotsos, "Indoor place recognition system for localization of mobile robots," in *2016 13th Conference on computer and robot vision (CRV)*, pp. 53–60, IEEE, 2016.

[130] N. Jacobs, N. Roman, and R. Pless, "Consistent temporal variations in many outdoor scenes," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 2007.

[131] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *European conference on computer vision*, pp. 304–317, Springer, 2008.

[132] M. Måns Larsson, E. Stenborg, L. Hammarstrand, M. Pollefeys, T. Sattler, and F. Kahl, "A cross-season correspondence dataset for robust semantic segmentation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9524–9534, 2019.

[133] M. Milford, "Vision-based place recognition: how low can you go?," *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 766–789, 2013.

[134] J. Mount and M. Milford, "2d visual place recognition for domestic service robots at night," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4822–4829, 2016.

[135] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, 2006.

[136] D. Bai, C. Wang, B. Zhang, X. Yi, and X. Yang, "Sequence searching with cnn features for robust and fast visual place recognition," *Computers & Graphics*, vol. 70, pp. 270–280, 2018.

[137] M. Magnusson, H. Andreasson, A. Nuchter, and A. J. Lilienthal, "Appearance-based loop detection from 3d laser data using the normal distributions transform," in *2009 IEEE International Conference on Robotics and Automation*, pp. 23–28, IEEE, 2009.

[138] S. Lowry and M. J. Milford, "Supervised and unsupervised linear learning techniques for visual place recognition in changing environments," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 600–613, 2016.

[139] M. S. Drew, J. Wei, and Z.-N. Li, "Illumination-invariant color object recognition via compressed chromaticity histograms of color-channel-normalized images," in *Sixth In-*

ternational Conference on Computer Vision (IEEE Cat. No. 98CH36271), pp. 533–540, IEEE, 1998.

[140] A. Ranganathan, S. Matsumoto, and D. Ilstrup, "Towards illumination invariance for visual localization," in *2013 IEEE International Conference on Robotics and Automation*, pp. 3791–3798, IEEE, 2013.

[141] O. Vysotska and C. Stachniss, "Lazy data association for image sequences matching under substantial appearance changes," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 213–220, 2015.

[142] O. Vysotska and C. Stachniss, "Effective visual place recognition using multi-sequence maps," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1730–1736, 2019.

[143] G. Hudson, A. Léger, B. Niss, I. Sebestyén, and J. Vaaben, "JPEG-1 standard 25 years: past, present, and future reasons for a success," *Journal of Electronic Imaging*, vol. 27, p. 040901, Aug. 2018. Publisher: SPIE.

[144] J. Hu, S. Song, and Y. Gong, "Comparative performance analysis of web image compression," in *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 1–5, 2017.

[145] D. Mateika and R. Martavičius, "Analysis of the compression ratio and quality in aerial images," *Aviation*, vol. 11, no. 4, pp. 24–28, 2007.

[146] R. F. Haines, *The effects of video compression on acceptability of images for monitoring life sciences experiments*, vol. 3239. National Aeronautics and Space Administration, Office of Management, 1992.

[147] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[148] V. Sze, M. Budagavi, and G. J. Sullivan, "High efficiency video coding (hevc)," in *Integrated circuit and systems, algorithms and architectures*, vol. 39, p. 40, Springer, 2014.