Exploiting
the Test-time Reference Map for Visual Place Recognition

Mubariz ZAFFAR

EXPLOITING THE TEST-TIME REFERENCE MAP FOR VISUAL PLACE RECOGNITION

EXPLOITING THE TEST-TIME REFERENCE MAP FOR VISUAL PLACE RECOGNITION

Dissertation

for the purpose of obtaining the degree of doctor at Delft University of Technology, by the authority of the Rector Magnificus Prof. dr. ir. H. Bijl,
chair of the Board for Doctorates, to be defended publicly on Wednesday, 6 May 2026 at 15:00

by

Mubariz ZAFFAR

Born in Dera Ismail Khan, Pakistan.

This dissertation has been approved by the

promotor: Dr. J.F.P. Kooij
copromotor: Dr. L. Nan

Composition of the promotion committee:

Rector Magnificus,	Chairperson
Dr. J.F.P. Kooij,	Delft University of Technology
Dr. L. Nan,	Delft University of Technology
Independent members:	
Prof. dr. M. Shah,	University of Central Florida, United States of America
Dr. N. Strisciuglio,	University of Twente, The Netherlands
Dr. J. Civera Sancho,	University of Zaragoza, Spain
Prof. dr. ir. M. Wisse,	Delft University of Technology
Prof. dr. ir. J. Hellendoorn,	Delft University of Technology (reserve member)

The work in the thesis has been supported by the TU Delft AI Labs & Talent Programme.

Keywords: Visual place recognition, localization, representation learning, feature extraction, description and matching.

Style: TU Delft House Style, with modifications by Moritz Beller
`https://github.com/Inventitech/phd-thesis-template`

The author set this thesis in LaTeX using the Libertinus and Inconsolata fonts.

ISBN: 978-94-6518-287-2

An electronic version of this dissertation is available at
`http://repository.tudelft.nl/`.

If the goal isn't visible yet, let the quest suffice for now. [Translated from Urdu]

Faiz Ahmed Faiz

نہیں نگاہ میں منزل تو جستجو ہی سہی

CONTENTS

SUMMARY

Visual Place Recognition (VPR) is a fundamental problem in computer vision and robotics, where the task is to recognize whether a place has been visited before using only visual input. It enables key capabilities such as loop closure in Simultaneous Localization and Mapping (SLAM), image-based localization, landmark detection/retrieval, and visual navigation. The traditional challenges in VPR include robustly matching images despite changes in viewpoint, illumination, season, and dynamic content. The rise of deep-learning has led to VPR methods that demonstrate significant robustness to these challenges. This thesis instead investigates three less studied but important research challenges in VPR: domain generalization, uncertainty estimation, and localization accuracy. It provides the key insight into how the test-time reference map, traditionally used only for retrieval, can be leveraged more actively to improve VPR across all three research challenges. The author argues that the information already present in the reference map can be systematically exploited to improve VPR performance without requiring additional sensors, supervision data, or retraining.

The thesis begins by situating VPR in the broader landscape of Visual-based Localization (VBL) and surveying its applications from loop closure to 3D reconstruction and map-based navigation. The motivation to study VPR as a central module that enables other types of VBL, and also its stand-alone applications, is presented. Next, the literature within VPR and VBL is thoroughly reviewed and categorized. A central observation is that the progress in VPR has been fragmented: robotics and vision communities use different datasets, metrics, and evaluation practices, producing an inconsistent picture of the state-of-the-art.

To address this, Chapter 3 presents VPR-Bench, a unified, open-source evaluation framework designed to converge the disparate practices across communities. VPR-Bench integrates the largest curated collection of techniques, datasets, and evaluation metrics to date (of publishing), re-implementing popular methods with consistent templates, dataset formats, and ground truths. Importantly, the framework extends beyond standard precision–recall metrics: it incorporates Receiver-Operator-Characteristic (ROC) analyses to evaluate true-negative detection (new-place recognition), quantifies viewpoint and illumination invariance using variation-annotated datasets, and exposes the effects of ground-truth manipulation on reported rankings. The chapter also provides a meta-analysis enabled by VPR-Bench: it examines descriptor size, CPU vs GPU runtime trade-offs, and the relationship between retrieval time, map size, and platform dynamics, and it compares how viewpoint variance vs invariance impacts different applications. By re-releasing code, datasets, and standardized ground truths, VPR-Bench establishes a common foundation for future VPR evaluations, helping to clarify which methods perform well under which conditions and why. Nevertheless, one of the key observations from this chapter is that there is no universally best VPR method.

Therefore, Chapter 4 tackles cross-domain robustness—how VPR methods generalize when training and testing environments differ. While recent Vision Foundation Model (VFM) backbones (e.g., transformer-based descriptors) yield very high recall on datasets similar

to their training distribution, performance degrades on visually distinct environments: a clear train-test domain gap exists. Leveraging the practicality that many VPR applications provide the reference map offline at test time, this chapter proposes Reference-Set Finetuning (RSF): a self-supervised finetuning strategy that uses the available test-time reference images (with poses) and augmentations to adapt models to the deployment domain. RSF reduces the domain mismatch with zero annotation cost and improves retrieval robustness on diverse testbeds. The chapter frames key questions: effectiveness on small finetuning sets, generalization after finetuning, and universality of the strategy across domains. It provides empirical evidence that the targeted use of test-time reference data can significantly reduce domain gaps while remaining complementary to stronger VFMs and large-scale data-driven training. However, even after domain adaptation, it is still possible that some images are just inherently too difficult to match, e.g., white walls across different unique places are indeed almost indistinguishable. A reliability estimation for VPR is thus needed.

To examine reliability Chapter 5 studies the task of uncertainty estimation in VPR. Perceptual aliasing creates aleatoric uncertainty—inherent ambiguity that cannot be eliminated by more data—leading to high-confidence false positives, which are dangerous in downstream tasks (e.g., erroneous loop closures). Existing uncertainty strategies fall into three categories: retrieval-based heuristics (descriptor distances/ratios), data-driven aleatoric uncertainty estimators, and geometric verification via local feature matching. None of these categories uses the pose information freely available in the reference map. This chapter thus introduces Spatial Uncertainty Estimation (SUE), a simple baseline that leverages map metadata: SUE measures the spatial spread of top-ranked reference poses. The underlying intuition here is that tight clusters indicate confident matches; dispersed poses indicate ambiguity. Compared against retrieval, data-driven, and geometric verification approaches, SUE provides highly efficient uncertainty estimates that outperform other lightweight methods and approach the reliability of geometric verification at a fraction of the cost. The chapter also studies how SUE complements expensive geometric verification, enabling designs that balance accuracy and runtime for safety-critical deployment. While domain-adapted, confidence-based VPR can help in pure image retrieval applications, it is unclear how VPR can relate to and benefit accurate localization.

Chapter 6 of this thesis thus addresses localization accuracy using VPR, highlighting a fundamental limitation of discrete reference maps: quantization error when queries fall between the anchor reference poses. To reduce this base error without collecting more images, employing a secondary Relative-Pose-Estimation (RPE) module or building full 3D reconstructions, this chapter proposes Continuous Place-descriptor Regression (CoPR)—densification in the feature space. CoPR regresses descriptors at novel poses using only existing anchor descriptors and pose relations, via both linear and non-linear models, enabling a denser, continuous map that a retrieval system can exploit. Experiments show that CoPR improves localization accuracy on multiple benchmarks, with the largest gains when densified descriptor maps are combined with viewpoint-variant VPR encoders, since viewpoint invariance inherently ignores spatial differences and limits densification benefits. The chapter contrasts interpolation and extrapolation strategies, demonstrates cases where RPE cannot recover without CoPR, and shows how the densified VPR reference maps complement coarse-to-fine localization to produce more accurate and reliable localization.

Collectively, this thesis reframes the reference map from a passive database to an active,

exploitable resource. By systematically using test-time map information—through finetuning (RSF), spatial uncertainty (SUE), and feature-space densification (CoPR)—it delivers measurable improvements in robustness, reliability, and localization accuracy without additional sensing or full reconstruction. These contributions advocate for map-aware VPR: systems that reason about map structure and uncertainty, adapt to deployment domains, and interpolate knowledge across the continuous feature space. Such systems will be better suited to real-world deployment in robotics and autonomous systems where reliability, adaptability, and precision are essential.

SAMENVATTING

Visual Place Recognition (VPR) is een fundamenteel probleem in computer vision en robotica, waarbij de taak bestaat uit het herkennen of een plaats eerder is bezocht op basis van uitsluitend visuele input. Het maakt belangrijke mogelijkheden mogelijk, zoals loop closure in Simultaneous Localization and Mapping (SLAM), beeldgebaseerde lokalisatie, landmarkdetectie/-retrieval en visuele navigatie. De traditionele uitdagingen in VPR omvatten het robuust matchen van beelden ondanks veranderingen in gezichtspunt, verlichting, seizoen en dynamische inhoud. De opkomst van deep learning heeft geleid tot VPR-methoden die aanzienlijke robuustheid tegen deze uitdagingen tonen. Dit proefschrift onderzoekt in plaats daarvan drie minder bestudeerde maar belangrijke onderzoeksvragen binnen VPR: domeingeneralisatie, onzekerheidsschatting en lokalisatienauwkeurigheid. Het biedt het belangrijke inzicht dat de referentiekaart tijdens testtijd—traditioneel alleen gebruikt voor retrieval—actiever kan worden ingezet om VPR te verbeteren op al deze drie onderzoeksuitdagingen. De auteur stelt dat de informatie die al aanwezig is in de referentiekaart systematisch kan worden benut om de VPR-prestaties te verbeteren zonder extra sensoren, supervisiedata of hertraining.

Het proefschrift opent met het positioneren van VPR binnen het bredere landschap van Visual-based Localization (VBL) en geeft een overzicht van toepassingen variërend van loop closure tot 3D-reconstructie en kaartgebaseerde navigatie. De motivatie om VPR te bestuderen als een centrale module die andere vormen van VBL mogelijk maakt—en ook zijn autonome toepassingen—wordt uiteengezet. Vervolgens wordt de literatuur binnen VPR en VBL uitgebreid besproken en geclassificeerd. Een centrale observatie is dat de vooruitgang binnen VPR gefragmenteerd is: robotica- en vision-gemeenschappen gebruiken verschillende datasets, metriek en evaluatiepraktijken, wat leidt tot een inconsistent beeld van de stand van zaken.

Om dit aan te pakken presenteert Hoofdstuk 3 VPR-Bench, een uniforme, open-source evaluatieomgeving ontworpen om de uiteenlopende praktijken binnen gemeenschappen te verenigen. VPR-Bench integreert de grootste samengestelde collectie van technieken, datasets en evaluatiemetrieken tot op de datum van publicatie, en herimplementeert populaire methoden met consistente sjablonen, datasetformaten en ground truths. Belangrijk is dat het framework verder gaat dan standaard precision–recall-metriek: het omvat ROC-analyses voor het evalueren van true-negative detectie (nieuwe-plaatsherkenning), kwantificeert viewpoint- en illumination-invariantie met behulp van variatie-geannoteerde datasets, en laat zien hoe manipulatie van ground truths de gerapporteerde rangschikkingen beïnvloedt. Het hoofdstuk biedt ook een meta-analyse mogelijk gemaakt door VPR-Bench: het onderzoekt descriptorgrootte, CPU- versus GPU-runtime-afwegingen, de relatie tussen retrievaltijd, kaartgrootte en platformdynamiek, en het vergelijkt hoe viewpoint-variantie versus -invariantie verschillende toepassingen beïnvloedt. Door code, datasets en gestandaardiseerde ground truths opnieuw vrij te geven, legt VPR-Bench een gemeenschappelijke basis voor toekomstige VPR-evaluaties en helpt het duidelijk te maken welke methoden goed presteren onder

welke omstandigheden en waarom. Niettemin is een van de belangrijkste observaties uit dit hoofdstuk dat er geen universeel beste VPR-methode bestaat.

Daarom behandelt Hoofdstuk 4 robuustheid over domeinen heen—hoe VPR-methoden generaliseren wanneer de trainings- en testomgevingen van elkaar verschillen. Hoewel recente Vision Foundation Model (VFM)-backbones (zoals transformer-gebaseerde descriptors) zeer hoge recall bieden op datasets die vergelijkbaar zijn met hun trainingsdistributie, verslechtert de prestatie sterk in visueel verschillende omgevingen: een duidelijk train-test domeingat bestaat. Met gebruikmaking van het praktische gegeven dat vele VPR-toepassingen de referentiekaart offline beschikbaar hebben tijdens testtijd, stelt dit hoofdstuk Reference-Set Finetuning (RSF) voor: een zelf-gecontroleerde finetuningstrategie die de beschikbare referentiebeelden (met poses) en augmentaties gebruikt om modellen aan te passen aan het inzetdomein. RSF verkleint het domeinverschil zonder annotatiekosten en verbetert de retrievalrobuustheid op diverse testomgevingen. Het hoofdstuk formuleert belangrijke vragen: effectiviteit met kleine finetuning-sets, generalisatie na finetuning en universaliteit van de strategie over domeinen heen. Het levert empirisch bewijs dat gerichte inzet van testtijd-referentiegegevens het domeingat aanzienlijk kan verkleinen, terwijl het complementair blijft aan sterkere VFM's en grotere trainingscorpora. Echter, zelfs na domeinadaptatie blijven sommige beelden intrinsiek moeilijk te matchen—bijvoorbeeld witte muren van verschillende unieke plaatsen zijn bijna niet van elkaar te onderscheiden. Daarom is een betrouwbaarheidsinschatting voor VPR noodzakelijk.

Om betrouwbaarheid te onderzoeken bestudeert Hoofdstuk 5 de taak van onzekerheidsschatting binnen VPR. Perceptuele aliassing creëert aleatorische onzekerheid—intrinsieke ambiguïteit die niet kan worden verwijderd met meer data—wat leidt tot hoog-zelfvertrouwen-false-positives die gevaarlijk zijn voor vervolgmodules (bijv. foutieve loop closures). Bestaande onzekerheidsstrategieën vallen in drie categorieën: retrieval-gebaseerde heuristieken (descriptorafstanden/-ratio's), datagedreven aleatorische-onzekerheidsmodellen en geometrische verificatie via lokale feature-matching. Dit hoofdstuk introduceert daarom Spatial Uncertainty Estimation (SUE), een eenvoudige baseline die gebruikmaakt van kaartmetadata: SUE meet de ruimtelijke spreiding van top-gerangschikte referentieposes. De onderliggende intuïtie is dat compacte clusters duiden op betrouwbare matches; verspreide poses duiden op ambiguïteit. Vergeleken met retrieval-, datagedreven en geometrische-verificatiemethoden levert SUE zeer efficiënte onzekerheidsschattingen die andere lichte methoden overtreffen en de betrouwbaarheid van geometrische verificatie benaderen tegen een fractie van de kosten. Het hoofdstuk onderzoekt ook hoe SUE dure geometrische verificatie aanvult, waardoor ontwerpen mogelijk worden die nauwkeurigheid en runtime in evenwicht brengen voor veiligheidskritische toepassingen. Hoewel domein-aangepaste, betrouwbaarheidsbewuste VPR nuttig is voor pure image-retrievaltoepassingen, is het onduidelijk hoe VPR kan bijdragen aan of profiteren van nauwkeurige lokalisatie.

Hoofdstuk 6 van dit proefschrift behandelt daarom lokalisatienauwkeurigheid binnen VPR en benadrukt een fundamentele beperking van discrete referentiekaarten: kwantisatiefouten wanneer querybeelden zich tussen de ankerposes bevinden. Om deze basale fout te reduceren zonder extra beelden te verzamelen, een secundaire Relative-Pose-Estimation (RPE)-module te gebruiken of volledige 3D-reconstructies te bouwen, stelt dit hoofdstuk Continuous Place-descriptor Regression (CoPR) voor—densificatie in de feature-ruimte. CoPR regresseert descriptors op nieuwe poses uitsluitend via bestaande ankerdescriptors en

poserelaties, met zowel lineaire als niet-lineaire modellen. Hierdoor ontstaat een dichtere, continue kaart die door een retrievalsysteem kan worden benut. Experimenten tonen aan dat CoPR de lokalisatienauwkeurigheid verbetert op meerdere benchmarks, met de grootste winst wanneer gedensificeerde descriptor-kaarten worden gecombineerd met viewpoint-variant VPR-encoders, aangezien viewpoint-invariantie ruimtelijke verschillen samenklapt en het voordeel van densificatie beperkt. Het hoofdstuk vergelijkt interpolatie- en extrapolatiestrategieën, toont gevallen waarin RPE niet kan herstellen zonder CoPR, en laat zien hoe gedensificeerde VPR-referentiekaarten coarse-to-fine lokalisatie aanvullen om nauwkeurigere en betrouwbaardere lokalisatie te bereiken.

Gezamenlijk herkadert dit proefschrift de referentiekaart van een passieve databron tot een actief, benutbaar hulpmiddel. Door systematisch gebruik te maken van testtijdinformatie uit de kaart—via finetuning (RSF), ruimtelijke onzekerheid (SUE) en densificatie in de feature-ruimte (CoPR)—levert het meetbare verbeteringen op in robuustheid, betrouwbaarheid en lokalisatienauwkeurigheid zonder extra sensoren of volledige reconstructie. Deze bijdragen pleiten voor kaartbewuste VPR: systemen die redeneren over kaartstructuur en onzekerheid, zich aanpassen aan inzetdomeinen en kennis interpoleren in de continue feature-ruimte. Zulke systemen zullen beter geschikt zijn voor reële inzet in robotica en autonome systemen, waar betrouwbaarheid, aanpasbaarheid en precisie essentieel zijn.

1

INTRODUCTION

This chapter is partly based on *Oscar de Groot, ..., Mubariz Zaffar, ..., Julian F. P. Kooij, et al. A Vehicle System for Navigating among Vulnerable Road Users Including Remote Operation. IEEE Intelligent Vehicles Symposium, 2025.* [1].

Author contributions: Mubariz Zaffar worked on the LiDAR-based localization stack, created several pointcloud and lanelet maps for the test-areas, and designed the first version of the conference poster. All other authors worked on their various respective system modules and technical writing for this large group project.

1

KNowing *where you are* is important. It helps you to recognize where you came from (*the prior journey*), to identify if this is where you want to be (*the milestone check*), to scheme where you want to go next (*the planning*), and in getting there (*the navigation*). This, of course, has a philosophical and perhaps even a highly-roboticist connotation, but in this thesis it relates to the ability of any autonomous agent to localize itself in the real world. These autonomous agents could be different embodiments of artificial intelligence (AI), such as self-driving cars, cleaning robots, agricultural robots, medical robots, construction robots, delivery robots, or even your Google Maps mobile application. For example, Fig.1.1 shows the latest advances in augmented reality for Google Maps, where a live camera feed is used for accurate localization and augmented with useful navigation cues [2]. Nevertheless, all of these autonomous agents present significant economic, societal, and environmental benefits in a world that is already facing declining working age population [3].

Figure 1.1: An exemplar application of accurate localization is the latest advances in Google Maps. A live camera feed is used to accurately localize the phone's position and orientation, formally the 6 Degrees-of-Freedom (6-DoF) pose, with respect to a map. This pose then enables augmenting the image with useful navigational guidance.

Central to all of these autonomous agents is their ability to accurately localize themselves, i.e., determine their pose (position and orientation) in a given world. Different sensors could be used to perceive the environment for localization, such as monocular cameras, stereo cameras, LiDAR, RADAR, tactile sensors, etc. Since monocular cameras are low-cost, passive, information-rich, and widely-adopted compared to other sensor types, they are paramount for scalable, real-world autonomous agents. This thesis thus focuses on camera-only methods of localization that are accurate, robust, and reliable.

A number of different formulations exist to use monocular cameras for localization, which have overtime become independent research fields. These formulations use the same type of input and output variables but differ in their treatment and assumptions of these variables; and have been categorized under the umbrella term *Visual-based Localization (VBL)* [4]. These formulations of VBL include Visual Place Recognition (VPR),

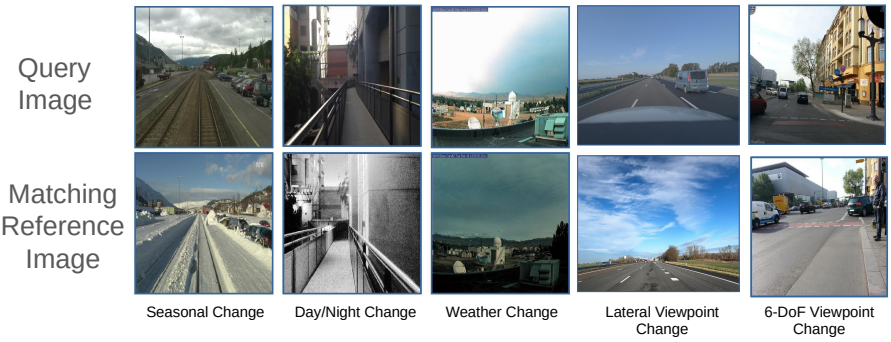


Figure 1.2: The challenge of matching places with robustness to viewpoint and appearance changes in Visual Place Recognition (VPR) is shown here.

Absolute-Pose-Estimation (APE), Relative-Pose-Estimation (RPE), Structure-based Localization (SBL), and Cross-view Localization (CVL), etc. They have corresponding strengths and weaknesses, as we will discuss later; however, one formulation, VPR, stands out because it is complementary to the other formulations. This complementarity of VPR will be expanded upon in the following subsections of this chapter, but is important to note at this stage.

Visual place recognition is essentially an image-retrieval formulation of visual-based localization, where given a query image of a place, the task is to retrieve a correct matching image of the same place from a reference database [1] (*the reference map*) with robustness to viewpoint and appearance changes [5]. Given the context of our application (i.e., real-world autonomous agents), this robustness is non-trivial, since changes in viewpoint, e.g., 6 Degrees-of-Freedom movement in handheld cameras and drones, and varying seasons, day-light conditions, weather patterns, could make the appearance/images of the same place look drastically different. This is demonstrated in Fig. 1.2. For most of the history of VPR, this has been the most important research challenge and has been tackled reasonably-well by the use of deep-learning-based models to encode viewpoint- and appearance-invariant representations of places, as evidenced by recent benchmarks and evaluations [6–9],.

Nevertheless, in addition to this challenge of varying appearances, there are three other key challenges in VPR that have received less attention and are of interest for this thesis. These include: a) deep-learning-based VPR methods may not generalize if the test environment is quite different from their training data, the commonly known train-test domain gap, b) images of different far-apart places could look highly similar due to repetitive patterns in the real-world, the so-called *perceptual-aliasing*, and c) localization using only VPR is not sub-meter level accurate since the best-matching reference image may still be several meters away from the query camera's location. This thesis studies all these three challenges in VPR and identified that the *test time reference map contains useful information to tackle them*.

[1] Please note that in thesis, and generally in literature, the terms *reference images*, *reference map*, *reference database*, and *reference set* have been used interchangeably, and refer to a set of images available *a priori* in a database with known camera poses.

1

The remainder of this chapter expands this introduction through dedicated sub-sections. Firstly, a motivation for studying VBL is provided along with its various categories. The central role of VPR for VBL is then described along with the applications and formal definition of VPR and its challenges. Next, the main research questions of this thesis are outlined, and a corresponding chapter outline is provided. Finally, the contributions of this thesis are enlisted.

1.1 WHY STUDY VISUAL-BASED LOCALIZATION?

This section first describes the need for accurate localization, then motivates the use of vision as a sensing modality for such localization, and lastly provides an overview of the various types of visual-based localization.

1.1.1 THE NEED FOR ACCURATE LOCALIZATION

There are a number of applications of accurate localization. In autonomous robotics, localization is central to the planning and navigation modules. Examples of such robots include cleaning robots, such as those from iRobot and Dyson; delivery robots, such as those developed by ServeRobotics and UberEats; self-driving cars, as being developed by Waymo, Zoox, Baidu, etc; and autonomous trucks, as researched by Aurora, Gatik, and Einride, etc. These autonomous robots present the potential to mitigate labour/driver shortages, improve quality of life, increase accessibility, and yield economical benefits.

Accurate localization also has benefits for augmented and virtual reality applications, which aim to improve user experience and quality of life. For example, recent advances in Google Maps use accurate localization to provide additional navigation cues [2]. Niantic uses accurate localization for immersive gaming experience, such as Pokemon-Go. Companies like Meta require accurate localization to realize Metaverse, an immersive 3D virtual world where users, represented by avatars, can interact socially and economically in real-time. Most applications of augmented and virtual reality are therefore based on accurate camera localization.

Creation and update of large-scale maps also requires accurate localization. Such maps are useful for navigation, e.g., more than 1 billion people use Google Maps every month [10, 11]. Companies like Footpath.ai aim to improve the accessibility of pedestrians and disabled people by mapping footpaths worldwide, and by providing insights to local government bodies to improving access for disabled people. Other mapping companies, such as Apple Maps, Bing Maps, Baidu Maps, TomTom, BeeMaps, etc, also require accurate localization to create Base and/or High-Definition (HD) maps.

1.1.2 MOTIVATION FOR VISUAL-BASED LOCALIZATION

Accurate and robust localization is needed to plan and navigate, and is thus crucial for autonomy. This autonomy could be related to simpler systems, such as household cleaning robots operating in low-risk, controlled environments, or more complex systems, such as robotaxis, that operate in high-risk rapidly-changing environments.

Different sensors could be used for localization, e.g., the most straightforward approaches include using either Global Navigation Satellite Systems (GNSS) together with an Inertial Measurement Unit (IMU) or LiDAR-based localization. Both of these have limitations.

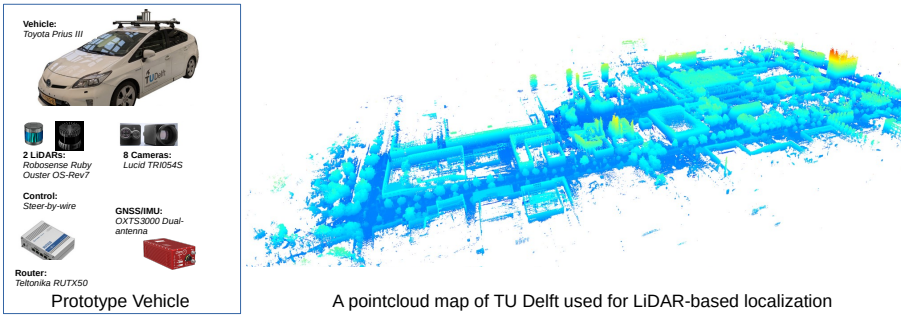


Figure 1.3: The demo vehicle for the Intelligent Vehicles Group at TU Delft and the pointcloud map created for the driverless demo at RSS 2025. [1]

Why not just use GNSS? Reliable localization using GNSS assumes direct line-of-sight to satellites, and hence, although it works quite well in open scenes, it fails to provide accurate localization in developed environments, such as urban canyons, city-centers, tunnels, and indoor environments [12]. GNSS is also prone to spoofing attacks[13], and accurate GNSS modules are expensive, e.g., several thousands of Euros [14].

Why not just use LiDAR? LiDAR is used by many autonomous systems currently for accurate localization, e.g., on a commercial side by Waymo robotaxis within their operational domain and on the academic side by the various teams that participated in the DARPA autonomous robotics challenges [15]. In fact, LiDAR is also used for localization by the author's research group, the Intelligent Vehicles Group (IVG), for the group's self-driving car, as demonstrated at RSS 2025 [1]. Fig. 1.3 shows the group's vehicle and the pointcloud map the author had created of TU Delft using a tuned version of LIO-SAM [16]. This pointcloud map was then used together with NDT-Scan Matching [17] for accurate LiDAR-based localization. While convenient for small demonstration areas, such LiDAR-based localization did not scale to large environments, as creating and updating such pointcloud maps requires extensive data collection, and is computationally and storage-wise inefficient. Moreover, like accurate GNNS receivers, LiDAR is also an expensive sensing modality: the LiDARs used on the IVG's self-driving car range between 15-25 thousand Euros.

Why use cameras? Unlike GNSS and LiDAR, monocular cameras are an inexpensive sensing modality. They provide semantically-rich information about the environment and are widely-adopted. More importantly, recording data with cameras is trivial and infact, a large amount of the human world has already been recorded multiple times by platforms such as Google-Street-View, Mapillary, BeeMaps, Footpath.ai, etc. Given that cameras are inexpensive and maps (monocular imagery) are either already available or easy-to-collect, an accurate camera-based localization system can truly unlock autonomy at scale. Thus, developing accurate, robust, and generalizable VBL frameworks is a useful research direction. VBL has, over the years, branched out into multiple categories.

1.1.3 CATEGORIES OF VISUAL-BASED LOCALIZATION

Visual-based localization is defined as retrieving the pose (position + orientation) as an output, given two inputs: a visual query material and a known space representation [4].

1

The term *visual query material* within the context of this thesis refers to a monocular colored/grayscale camera image, and the *known space representation* is some form of the post-processed *reference map*. Given this definition, different assumptions regarding the inputs and outputs led to different categories of VBL. Fig. 1.4 presents an overview of the various categories of VBL, which are overviewed in the following, and thoroughly reviewed in the next chapter.

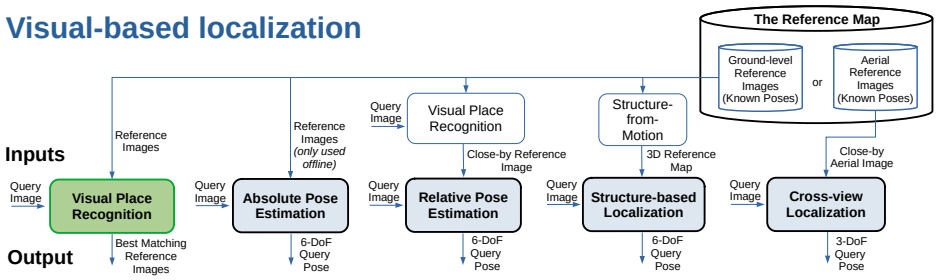


Figure 1.4: An overview of the different categories of Visual-based Localization (VBL) is provided. All of these formulations require a query image at test-time and use the reference map directly or indirectly. While the other approaches directly output the query camera pose, visual place recognition only provides a set of nearby reference images as an output. The query pose could then be approximated using the poses of the best matching reference images or by using relative pose estimation as a secondary stage.

VISUAL PLACE RECOGNITION

Visual Place Recognition (VPR) aims to retrieve a set of best-matching reference images as an output given an input query image and a reference map of images with known poses. The query camera pose can then be approximated given the poses of the best-matching reference images. Fig. 1.5 presents a block diagram of a VPR system. Generally, VPR is formulated as a representation learning problem, and has hence benefited from recent advances in deep representation learning [18]. It is considered to generalize across environments with no (or little) need for re-training/finetuning.

Assumptions: VPR assumes that the query camera pose is not accurately needed. Thus, localization using VPR alone is generally not accurate, and requires subsequent stages, such as relative pose estimation. The next section 1.2 will formally formulate VPR, discuss its applications and challenges but for now, we must briefly look at the other categories of VBL.

ABSOLUTE POSE ESTIMATION

Absolute Pose Estimation (APE) is the problem of regressing as an output a 6-DoF camera pose directly given an input query image. It is generally considered to be accurate and robust, and benefits from training on large-scale data with multiple appearances of places for robustness [19, 20].

Assumptions: APE assumes that the reference map containing images with labeled poses is available offline and is stored within the weights of a deep neural network at training time, and that a test-time query must come from a location available in reference map. Thus, APE methods require re-training for new environments, and do not generalize.

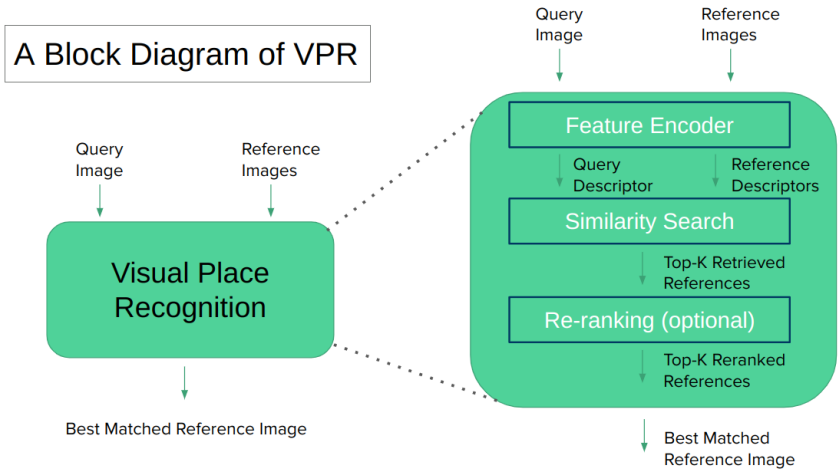


Figure 1.5: A block diagram of Visual Place Recognition (VPR). The query and reference images are encoded into feature vectors, a similarity search is performed (e.g., K-nearest-neigbour based with cosine distance), optionally the top-K retrieved reference are re-ranked based on additional filtering, and the nearest neighbor reference image is considered as the best matching reference image for the given query image.

RELATIVE POSE ESTIMATION

Relative Pose Estimation (RPE) is the problem of regressing a 6-DoF relative pose as an output between two input images: a query image and an anchor reference image looking at the same scene/place [21]. The pose for the anchor image is generally assumed to be known and, thus, given the relative pose from RPE, the absolute pose can be computed. The anchor reference image is determined given a weak-GPS prior, or through visual place recognition. RPE is generally considered to be accurate, especially methods using local feature matching [22].

Assumptions: RPE assumes that the anchor reference image has some visual overlap with the query image. Therefore, in situations where there is no ground-truth overlap between the query and anchor image, the pose output by RPE is arbitrary. It thus relies heavily on the quality of the mechanism used for mining the anchor image.

STRUCTURE-BASED LOCALIZATION

Structure-based localization estimates the 6-DoF camera pose as an output given two inputs: a query image and a 3D reference map of the environment. The 3D reference map is usually created by running structure-from-motion on the reference images. Among all the categories of VBL, structure-based localization is considered the most accurate.

Assumptions: Structure-based localization assumes that the reference map is available as a 3D pointcloud. This therefore requires running computationally-intensive pipelines on the reference images to first create a 3D map, which would then be used for localization. Creating such maps is expensive and does not scale well in large scenes, especially scenes with repetitive or non-salient features [20].

CROSS-VIEW LOCALIZATION

Cross-View Localization (CVL) aims to estimate a 3-DoF camera pose as an output given two inputs: a query image and an anchor reference aerial image of the same location [23, 24]. Because aerial images are geo-registered, knowing the location of a query in an aerial image automatically gives the pose as an output. The primary strength of CVL is that the reference aerial images for many countries are readily available and continuously updated.

Assumptions: CVL assumes that a reference anchor aerial image is given for a query image, based on either a weak-GPS prior or cross-view image retrieval. If the anchor aerial image does not depict the same visual content as the query, CVL outputs an arbitrary pose. Moreover, the assumption that aerial images are readily available does not hold for some countries.

All of these various categories of VBL have been thoroughly researched in existing literature and have corresponding assumptions, strengths, and weaknesses. Among these various categories, VPR is the formulation of interest in this thesis. The next section motivates this choice through VPR's standalone applications and complementarity to other VBL methods.

1.2 VPR: A CENTRAL MODULE?

This section presents the stand-alone applications of VPR and then within the context of other VBL formulations. It then presents the open research challenges in VPR.

1.2.1 THE CENTRAL ROLE OF VPR

VPR has stand-alone applications but also plays a central role to either enable or improve other types of VBL. First, the applications of VPR as a stand-alone task in computer vision are presented, and then its contributions to other types of VBL are presented.

APPLICATIONS OF VPR

A primary application of VPR is in loop closure for Simultaneous Localization and Mapping (SLAM), where the task is to create a map of an unknown environment and simultaneously localize itself within it. Loop-closure is critical to resolve the odometry drift in SLAM [25]. SLAM has benefits through out the spectrum of autonomous robotics, e.g., trivial household cleaning robots require first creating a map before they are operational. SLAM also enables mapping of non-trivial environments which are generally difficult to reach for humans and present unfriendly operating conditions, e.g., caves and mines [15].

VPR has applications for image cataloging, e.g., it can help to automatically categorize images of places from single or multiple users. An application of this is platforms such as Flickr, ImageShack, and Google Photos. It also has applications for image retrieval, for example, Google/Bing image search and Pinterest.

Navigation applications such Google-, TomTom-, and Apple-Maps also require VPR since commodity phone GPS is often inaccurate especially in cities and urban canyons. VPR can help resolve ambiguities in navigation when diverging lanes are encountered on highways or in cases of multi-level highway interchanges, a common challenge in China for GPS-based navigation.

VPR can help create 3D models of places from crowd-sourced imagery without requiring pose estimates for these images, such as the flagship *Building Rome in a Day* project [26]

and other follow-up works in this domain. Structure-from-Motion (SfM) pipelines like ColMap [27] require multiple images of the same place to create a 3D model, which can be achieved by running VPR on any collection of images.

Robotics also benefits from VPR, e.g., one application is in SLAM for robotics. In case of multi-robot applications, VPR could be used to stitch maps from different robots together in a single unified map [28]. Moreover, it can be used to resolve the kidnapped-robot-problem and the wake-up robot problem in autonomous robotics, where the robot is either carried to an unknown location or is uninitialized, respectively [5].

BENEFIT OF VPR FOR OTHER TYPES OF VBL

Besides the stand-alone applications of VPR, it also directly or indirectly benefits other types of VBL. This motivates that improving VPR is not just useful for its standalone applications but also for the applications of other VBL methods.

VPR for Absolute Pose Estimation: The primary strength of APE is the ability to directly regress the pose of a query image without requiring a search within a map at test-time. However, since APE is a learning-based method, it is limited by the training distribution it has seen. Even if the test data comes from the same environment as seen at training time, the nature of pose change or the appearance of a place could represent a different data distribution. This distribution shift could hurt the performance of APE systems, as studied for varying viewpoints by Sattler et al. [29]. VPR can help augment the training data of APE methods by retrieving images of the same place with varying viewpoints and appearances from crowd-sourced imagery, such as Mapillary.

VPR for relative pose estimation: VPR is a bottleneck for RPE methods. Localization using RPE assumes that an anchor image with overlapping content as that of the query image is available [21, 22, 30]. If the anchor image does not contain the same visual content as that of the query image, RPE methods output an arbitrary pose. Thus, improving VPR directly benefits RPE methods.

VPR for structure-based localization: Structure-based localization assumes that the reference map is available as a 3D structure [31]. This 3D structure is usually a pointcloud with associated local feature descriptors. The availability of such a 3D structure is possible if the reference data was collected through an RGB-D camera or if SfM was used to create 3D representation of the reference map. The later requires multiple images of the same place, which is achieved through VPR. Moreover, some structure-based localization methods use VPR to provide a coarse location estimate which is then refined further using 2D-to-3D feature matching [32, 33].

VPR for cross-view localization: CVL aims to regress the 3-DoF pose (2D location + 1D orientation) of a query image in a reference aerial image. This reference aerial image generally represents a large area, e.g., 200×200 meters, and thus a coarse estimate of the query location is enough to retrieve a relevant aerial image for this query. The coarse pose estimate is assumed available given a weak-GPS prior. VPR in its standard definition is not beneficial for CVL, since the VPR reference map is considered to contain ground-level

images and not aerial images. However, if the reference map in VPR would be a set of aerial images, VPR can be used to provide the coarse pose estimate for subsequent CVL. Such a formulation of VPR where reference images are aerial images is indeed a research task, known as cross-view image retrieval [34].

1.2.2 CHALLENGES IN VPR

The foremost challenges in VPR include robustness to viewpoint and appearance changes between the query and reference images, as discussed and presented earlier in Fig. 1.2. This section introduces other challenges in VPR that form the basis for this thesis.

LACK OF STANDARDIZED BENCHMARKS

Major advances have been made in VPR research, both in the computer vision and robotics communities. The computer vision community, in fact, refers to VPR as image-retrieval. [2] Both communities use different evaluation metrics, different datasets, and often different baselines. This lack of standardization leads to a lack of clarity regarding the state-of-the-art method, the useful datasets to report VPR performance, and the evaluation metrics we should care about. It is also unclear whether the two communities are trying to solve different challenges, for example, the need for viewpoint invariance.

DEGENERALIZATION WITH TRAIN-TEST DOMAIN GAP

A VPR method, by definition, is expected to generalize to new environments. However, new environments could present a data distribution different from the training data in VPR, and thus, the features learned at training time may not be useful for the environment seen at test time. This train-test domain gap is not just a problem for VPR but essentially for any learning-based approach. The standard solution in VPR to handle the train-test domain gap is to train deep-learning-based methods on large-scale datasets with varying appearances, thus broadening the data distribution. For example, the largest training dataset in VPR (GSV-Cities [35]) is consciously created from many cities worldwide to maximize diversity. In addition to this, the use of general-purpose feature extractors, such as Vision-Foundation-Models (VFMs), is expected to improve VPR generalization. Nevertheless, as this thesis will show later, VFM-based VPR methods trained on large-scale diverse datasets still suffer from train-test domain gap.

POOR UNCERTAINTY ESTIMATION IN VPR

The primary focus in VPR research has been to improve the robustness of VPR methods by learning robust feature spaces. As a result, uncertainty in VPR is usually modeled by the distance in feature space; the further two images are in the feature space, the lower the confidence that these two images belong to the same place. However, some images may inherently be too difficult to disambiguate. For example, two images of white walls in different offices will appear quite similar in the image/feature space. Such is the case for images of trees and essentially any features that appear similar across different places in the world. Handling this ambiguity requires good uncertainty estimation to reject false-positives. Rejecting such false-positives is critical for applications where VPR is used as a correction mechanism. For example, loop-closure in SLAM requires VPR for correcting

[2] Or perhaps the robotics community refers to image retrieval as VPR.

the localization estimate when the odometry becomes unreliable. Similarly, for map-based navigation, when the GPS fails to provide a reliable location estimate, VPR is used to verify the location estimate. Nevertheless, despite the significant need for good-quality uncertainty estimation in VPR, it remains an under-studied topic.

LOW ACCURACY OF VPR-ONLY LOCALIZATION

Localization is often motivated as an important application of VPR. However, the definition of VPR regulates that VPR should be viewpoint-invariant [5]: feature descriptors for different viewpoints of the same place should be the same. This disconnect, therefore, leads to only coarse localization with VPR and necessitates a two-stage formulation for accurate localization. First, an image in the neighbourhood of the query is retrieved using VPR, and a second stage of relative-pose-estimation follows this to refine the pose estimate. Such a formulation is prone to failures, since an incorrect retrieval with VPR cannot be resolved with RPE. Moreover, while VPR methods are known to generalize across different environments, RPE methods do not generalize well to new environments. A possibility for VPR-only accurate localization remains an open research challenge.

1.3 RESEARCH QUESTIONS AND CHAPTER OUTLINE

1.3.1 RESEARCH QUESTIONS

Given the challenges identified in VPR, the main research question guiding this thesis is:

Main Research Question

MQ: How to achieve generalizability, confidence, and accuracy using VPR for localization?

The underlying *key hypothesis* to investigate this main research question is that the *test-time reference map* contains useful cues that can help find answers to this question. The offline availability of the test-time reference map is the key assumption made in this thesis, and sets the application scope.

The main research question is further split into several sub-research questions to help dissect and study the problem. Naturally, it is first important to verify whether the existing methods perhaps already solve the problems of interest, i.e., generalization, uncertainty estimation, and accurate localization for VPR. Hence, the first sub-research question is:

SQ1: Are the existing methods in VPR already generalizable, accurate, and uncertainty-aware?

With a slight leak of answer to SQ1 already, the next sub-research question aims to address the problem of train-test domain generalization in VPR. It notes that the test-time reference map is actually available offline in VPR, and could perhaps contain useful information to bridge the domain gap. Following this thought, the next question is:

1

SQ2: How can the test-time reference map be used to bridge the train-test domain gap in VPR?

After investigating the role of train-test domain gap in VPR, this thesis investigates uncertainty estimation for VPR. An important observation here is that the test time reference map also contain associated poses. That is, once a set of best-matching reference images is retrieved for a given query, there is also an additional cue about this retrieval: the spatial spread of the best-matching reference images. The next sub-research question is thus:

SQ3: Could the poses in the test-time reference map and the spatial spread of the best-matching reference images help estimate uncertainty in VPR?

After investigating generalization and uncertainty estimation in VPR, this thesis studies the problem of accurate localization in VPR. If, theoretically, the reference images in VPR were densely collected, such that the best-matching reference image is spatially close to the query, then VPR-based retrieval could already provide an accurate localization estimate. Therefore, we investigate the sub-research question:

SQ4: Could we design strategies to densify reference maps in VPR that lead to accurate VPR-only localization?

1.3.2 CHAPTER OUTLINE

The research questions identified in the previous section are investigated as dedicated chapters in this thesis. Firstly, **Chapter 2** presents a detailed literature review of Visual-based Localization (VBL) and Visual Place Recognition (VPR). The different works within VBL are reviewed. A thorough overview of the various handcrafted and machine learning-based VPR methods is provided, along with the recent foundation models-based methods.

Chapter 3 studies the challenge of train-test domain gap in VPR. It is identified that even the SOTA VPR methods based on Vision-Foundation-Models (VFMs) and trained on large-scale diverse datasets still suffer from domain gaps. To bridge this domain gap, a new Reference-Set-Finetuning (RSF) strategy is proposed. The chapter then showcases how RSF can help bridge domain gaps in VPR by using only a small number of images for finetuning. The benefits of RSF for different VPR techniques and test domains are demonstrated.

Chapter 4 investigates the different uncertainty estimation methods in VPR. It is identified that existing uncertainty estimation methods could be classified into multiple categories, and that none of these categories exploit the useful information available in the reference map, i.e., the poses of the reference images. A new uncertainty estimation method is proposed, namely, Spatial-Uncertainty-Estimation (SUE), that uses the spatial spread of best-matching reference images as a proxy for VPR uncertainty. SUE is shown to outperform the other methods, and is also demonstrated to be complementary.

Chapter 5 discusses the inaccuracy of localization using VPR due to the inherent viewpoint-invariance requirement for VPR methods. It proposes that if the reference map was densely collected and VPR methods were trained to be viewpoint-variant, then VPR-

only localization could, in fact, provide accurate localization estimates. However, since densely collecting reference images is not practically possible, this chapter presents a new strategy to regress VPR feature descriptors at novel views. This new strategy for Continuous-Place-Descriptor-Regression (CoPR) is shown to significantly improve VPR-only localization accuracy. CoPR is also demonstrated to benefit Relative-Pose-Estimation using self-constructed examples.

Finally, **Chapter 6** concludes this thesis by reporting the key findings and concretely answering the research questions of this thesis. It provides a broader discussion on the results and analysis of the preceding chapters and identifies useful future directions. The author then presents his wishful thinking to end this thesis.

1.4 THESIS CONTRIBUTIONS

The core contribution of this thesis is the evidence of the significant benefits of the test-time reference map for three key requirements in VPR: generalization, uncertainty estimation, and localization accuracy. The reference map had been previously underutilized in VPR, and this thesis showcases how simple methods can exploit the reference map to help fulfill these three requirements. The concrete contributions are outlined below:

Benchmarking existing VPR methods to investigate generalization and accuracy: VPR is motvated for various applications, such as localization, loop closure in SLAM, image retrieval, and is a critical component of many autonomous navigation systems ranging from autonomous vehicles to drones and computer vision systems. While the concept of place recognition has been around for many years, VPR research has grown rapidly as a field over the past decade due to improving camera hardware and its potential for deep learning-based techniques, and has become a widely studied topic in both the computer vision and robotics communities. This growth, however, has led to fragmentation and a lack of standardisation in the field, especially concerning performance evaluation. Moreover, the notion of viewpoint and illumination invariance of VPR techniques has largely been assessed qualitatively and hence ambiguously in the past. Chapter 3 of this thesis addresses these gaps through a new comprehensive open-source framework for assessing the performance of VPR techniques, dubbed "VPR-Bench". VPR-Bench[3] introduces two much-needed capabilities for VPR researchers: firstly, it contains a benchmark of 12 fully-integrated datasets and 10 VPR techniques, and secondly, it integrates a comprehensive variation-quantified dataset for quantifying viewpoint and illumination invariance. This chapter analyzes popular evaluation metrics for VPR from both the computer vision and robotics communities, and discusses how these different metrics complement and/or replace each other, depending upon the underlying applications and system requirements. The analysis reveals that no universal SOTA VPR technique exists, since: (a) state-of-the-art (SOTA) performance is achieved by 8 out of the 10 techniques on at least one dataset, (b) SOTA technique in one community does not necessarily yield SOTA performance in the other, given the differences in datasets and metrics. Furthermore, this chapter identifies key open challenges since: (c) all 10 techniques suffer greatly in perceptually-aliased and less-structured environments, (d) all techniques suffer from viewpoint variance where lateral change has less effect than 3D change, and

[3]Open-sourced at: `https://github.com/MubarizZaffar/VPR-Bench`

1

(e) directional illumination change has more adverse effects on matching confidence than uniform illumination change. This chapter also presents detailed meta-analyses regarding the roles of varying ground-truths, platforms, application requirements, and technique parameters. Finally, VPR-Bench provides a unified implementation to deploy these VPR techniques, metrics and datasets, and is extensible through templates.

The content in this chapter is published in the International Journal of Computer Vision (IJCV 2021). [8]

Designing a finetuning strategy given test-time reference map to bridge train-test domain gap: Visual Place Recognition (VPR) has been studied heavily as the task of retrieving an image of the same place with a focus on robustness to viewpoint and appearance changes. Recent works show that some VPR benchmarks containing drastic viewpoint and appearance changes are solved by methods using Vision-Foundation- Model backbones and trained on large-scale and diverse VPR-specific datasets. Several benchmarks remain challenging, particularly when the test environments differ significantly from the usual VPR training datasets. Chapter 4 proposes a complementary, unexplored source of information to bridge the train-test domain gap, which can further improve the performance of State-of-the-Art (SOTA) VPR methods on such challenging benchmarks. Concretely, this chapter identifies that the test-time reference set, the "map", contains images and poses of the target domain and must be available before the test-time query is received in several VPR applications. Therefore, we propose to perform simple Reference- Set-Finetuning (RSF) of VPR models on the map, boosting the SOTA ($\approx 2.3\%$ increase on average for Recall@1) on these challenging datasets. Finetuned models retain generalization, and RSF works across diverse test datasets.

The content in this chapter is published in the proceedings of the IEEE International Conference on Computer Vision (ICCV 2025) [36] and was presented as a poster at the Large-scale Cross-device Localization Workshop.

Formulating a new uncertainty estimation method for VPR: In Visual Place Recognition (VPR), the pose of a query image is estimated by comparing the image to a map of reference images with known reference poses. As is typical for image retrieval problems, a feature extractor maps the query and reference images to a feature space where a nearest neighbor search is then performed. However, till recently, little attention has been given to quantifying the confidence that a retrieved reference image is a correct match. Highly certain but incorrect retrieval can lead to catastrophic failure of VPR-based localization pipelines. Chapter 5 compares for the first time the main approaches for estimating the image-matching uncertainty, including the traditional retrieval-based uncertainty estimation and more recent data-driven aleatoric uncertainty estimation, and the computationally intensive geometric verification. This chapter further formulates a simple baseline method, SUE[4], which, unlike the other methods, considers the freely available poses of the reference images in the map. The experiments in this chapter reveal that a simple L2-distance between the query and reference descriptors is already a better estimate of image-matching uncertainty than current data-driven approaches. SUE outperforms the other efficient uncertainty estimation

[4]Open-sourced at: `https://github.com/MubarizZaffar/SUE`

methods, and its uncertainty estimates complement the computationally expensive geometric verification approach.

The content in this chapter is published in the proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2024) and was presented as a highlight poster at the main conference. [37]

Regressing feature descriptors at unseen novel views to densify reference maps in VPR: Visual Place Recognition (VPR) is often motivated as an image-based localization method that estimates the camera location of a query image by retrieving the most similar reference image from a map of geo-tagged reference images. Chapter 6 looks into two fundamental bottlenecks for VPR's localization accuracy: reference map sparseness and viewpoint invariance. Firstly, the reference images for VPR are only available at sparse poses in a map, which enforces an upper bound on the maximum achievable localization accuracy through VPR. This chapter therefore, proposes Continuous Place-descriptor Regression (CoPR) to densify the map and improve localization accuracy. It further studies various interpolation and extrapolation models to regress additional VPR feature descriptors from only the existing references. Secondly, this chapter compares different feature encoders and shows that CoPR presents value for all of them. The trained CoPR models are evaluated on three existing public datasets and report on-average around 30% improvement in VPR-based localization accuracy using CoPR, on top of the 15% increase by using a viewpoint-variant loss for the feature encoder. The complementary relation between CoPR and Relative Pose Estimation is also discussed.

The content in this chapter is published in the IEEE Transactions on Robotics (T-RO 2023). [38]

2

Related Work

The previous chapter has outlined several different categories of visual-based localization. A modular overview of these categories was presented in Fig. 1.4. In this chapter, the literature in these categories is reviewed in detail, starting with our category of interest: Visual Place Recognition. An overview of the categorization of the reviewed literature in this chapter is shown in Fig 2.1.

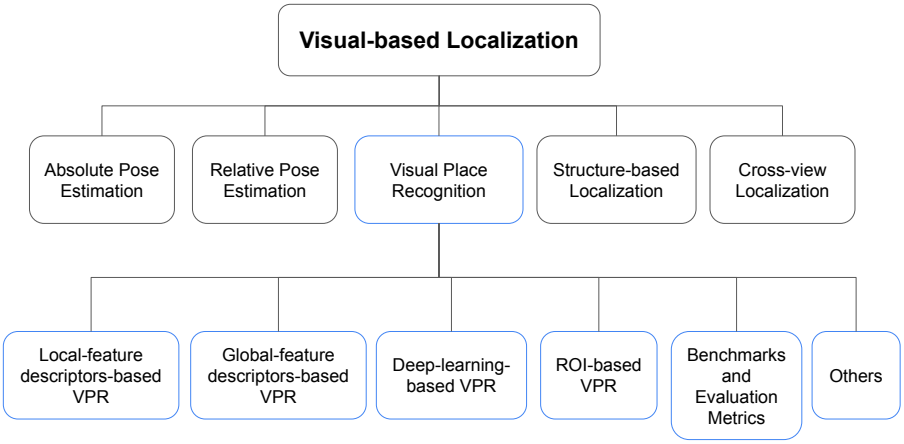


Figure 2.1: An overview of the literature reviewed and categorized in this chapter. More depth is dedicated to VPR, since that is the focus of this thesis.

2.1 VISUAL PLACE RECOGNITION

The existing literature in VPR can largely be broken down into: 1) Handcrafted feature descriptors-based VPR techniques, 2) Deep-learning-based VPR techniques, 3) Regions-of-Interest-based VPR techniques. All of these major classes have their trade-offs between matching performance and computational requirements.

2.1.1 LOCAL FEATURE DESCRIPTORS-BASED VPR

Handcrafted feature descriptors can be further sub-divided into two major classes: local feature descriptors and global feature descriptors. The most popular local feature descriptors developed in the vision community include Scale Invariant Feature Transform (SIFT [39]) and Speeded Up Robust Features (SURF [40]). These descriptors have been used for the VPR problem by [41], [42], [43], [44] and [45]. A probabilistic visual-SLAM algorithm was presented by [46], namely Frequent Appearance-based Mapping (FAB-MAP), that used SURF as the feature detector/descriptor and represented places as visual words. Odometry information was integrated into FAB-MAP by [47] to achieve Continuous Appearance Trajectory-based SLAM (CAT-SLAM) using a Rao–Blackwellised particle filter. CenSurE (Center Surround Extremas by [48]) is another popular local feature descriptor which has been used for VPR by [49]. FAST ([50]) is a popular high-speed corner detector that has been

used in combination with the SIFT descriptor for SLAM by [51]. Matching of local feature descriptors is a computationally intense process which has been addressed by the Bag of visual Words (BoW [52]) approach. BoW collects visually similar features in dedicated bins (pre-defined or learned by training a visual-dictionary) without topological consideration, enabling direct matching of BoW descriptors. Some of the techniques using BoW for VPR include the works of [53], [54], [55], and [56]. [57] presents a new methodology to estimate the distinctiveness of local feature descriptors in a query image from closely related matches in reference descriptor space, thereby utilising salient features within the image. While the hand-crafted local features like SIFT and SURF had been widely used for VPR, recent advances include learnt local features, for example, LIFT [58], R2D2 [59], SuperPoint [60], D2-net [61]. DELF [62] is a deep-learning-based local feature extractor and descriptor, that is used with geometric verification for large-scale image retrieval. Mast3R [22] is primarily a learning-based pose-estimation method but it can also be used as a local feature extractor and descriptor. Conclusively, while local feature descriptors generally lead to accurate VPR, they are computationally expensive and have high storage requirements.

2.1.2 GLOBAL FEATURE DESCRIPTORS-BASED VPR

Global feature descriptors create a holistic signature for an entire image, and Gist ([63]) is one of the most popular global feature descriptor. Working on panoramic images, [64], [65] used Gist for VPR. [66] combined Gist with BRIEF ([67]) to perform large scale visual-SLAM. [68] used Whole-Image SURF (WI-SURF), which is a global variant of SURF to perform place recognition. Operating on sequences of raw RGB-images, Seq-SLAM ([69]) uses global normalized pixel-intensity matching to perform VPR in challenging conditionally-variant environments. The original Seq-SLAM algorithm assumes constant speed of the robotic platform, thus, [70] extended Seq-SLAM to consider variable speed instead. [71] extract scene signatures from an image by utilising some *a priori* environment information and describe them using HOG-descriptors. DenseVLAD presented by [72] is a Vector-of-Locally-Aggregated-Descriptors-based approach using densely sampled SIFT keypoints, which has been shown to perform similar to deep-learning-based techniques in [73] [74]. A more recent usage of traditional handcrafted feature descriptors for VPR was presented in CoHOG ([75]) by the author of this thesis, which focuses on entropy-rich regions in an image and uses HOG as the regional descriptor for convolutional-regional matching. Global feature descriptors have significantly lower computational needs than local feature descriptors, but tend to be less accurate.

2.1.3 DEEP LEARNING-BASED VPR

Similar to other domains of computer vision, deep-learning and especially Convolutional-Neural-Networks (CNNs) are a game-changer for the VPR problem by achieving unprecedented invariance to conditional changes. By employing off-the-shelf pre-trained neural nets, [76] used features from the Overfeat Network ([77]) and combined them with the spatial filtering scheme of Seq-SLAM. This work was followed up by [78], where two neural networks (namely AMOSNet and HybridNet) were trained specifically for VPR on the Specific Places Dataset (SPED). AMOSNet was trained from scratch on SPED, while the weights for HybridNet were initialised from the top-5 convolutional layers of Caffe-Net ([79]). An end-to-end neural-network-based holistic descriptor NetVLAD is introduced by [80], where

a new VLAD (Vector-of-Locally-Aggregated-Descriptors ([81])) layer is integrated into the CNN architecture, achieving excellent place recognition results. A convolutional auto-encoder network is trained in an unsupervised fashion by [82], utilizing HOG-descriptors of images and synthetic viewpoint variations for training. The work of [62] was extended to DELG (DEep Local and Global Features by [83]) combining generalized mean pooling for global descriptors and attention mechanism for local features. [84] presented that state-of-the-art image-retrieval performance can be achieved by mining local features from CNN activation tensors and by performing spatial verification on these channel-wise local features, which can be then converted into global image signatures by using Bag-of-Words description. The work of [85] (GeM) introduces a new trainable 'Generalised Mean' layer into the deep image-retrieval architecture which has been shown to provide a performance boost. [86] draw their inspiration from brain architectures of fruit flies, train a sparse two-layer neural-network and combined it with Continuous-Attractor-Networks to summarise temporal information.

The current ongoing trend in the computer vision community is to learn universal and generalizable feature extractors, aka, vision foundation models. This has gained significant traction recently, after the release of the DinoV2 [87] Vision-Foundation Model (VFM), and other such foundation models [88, 89]. Naturally, this was picked up by the VPR and image retrieval community. Anyloc [90] investigated using DinoV2 as an off-the-shelf feature extractor. Many concurrent works subsequently showed that the performance benefits are significantly larger when DinoV2 is finetuned on VPR-specific data and training objectives [6, 7, 91, 92]. CricaVPR [7] proposes to use correlation between images in the batch with feature aggregation at multiple scales to produce robust global features. SALAD [91] uses the Sinkhorn algorithm to aggregate the global and local DinoV2 tokens for VPR. Authors of SelaVPR [92] add serial and parallel adapters to the DinoV2 architecture. Finally, BoQ [6] proposes to learn queries useful for VPR using the attention mechanism of transformers, and demonstrates that these learnable queries work with both older (ResNet) and newer (DinoV2) feature extraction backbones. Deep-learning-based feature descriptors have significantly outperformed handcrafted feature descriptors for VPR, but generally entail much higher feature encoding time.

2.1.4 REGIONS-OF-INTEREST-FOCUSED VPR

Prior to the attention mechanism, researchers used Regions-of-Interest (ROIs) to introduce the concept of salience into VPR, and to ensure that static, informative, and distinct regions are used for place recognition. Regions of Maximum Activated Convolutions (R-MAC) are used by [93], where max-pooling across cropped areas in CNN layers' features define/extract ROIs. This work on R-MAC is further advanced by [94], where a Siamese Network is trained with a Triplet loss on the Landmarks dataset ([95]). However, [96] argues that ranking-based loss functions (image-pairs, triplet-loss, n-tuples, etc.) are not optimal for the final task of achieving higher mAP and therefore propose a new ranking-loss that directly optimizes mAP. This mAP-based ranking loss function, in combination with GeM, achieves state-of-the-art retrieval performance. High-level features encoded in earlier neural-network layers are used for region-extraction, and the following low-level features in later layers are used for describing these regions in the work of [97]. This work is then followed up with a flexible attention-based model for region extraction by [98]. [99] draw their inspiration from

NetVLAD and R-MAC, thereby combining VLAD description with ROI-extraction to show significant robustness to appearance- and viewpoint-variation. Photometric-normalisation using both handcrafted and learning-based methodology is investigated by [100] to achieve illumination-invariance for place recognition.

It is relevant to note that explicit ROI extraction is not a requirement for salient VPR, since salient regions could also be learned implicitly, as was shown early on by Chen et al [78]. Nevertheless, through the rise of vision-transformers and the attention mechanism within them, explicit ROI extraction has been replaced by implicitly learning to combine/discard information from various image patches. For example, Bag-of-Queries [6] and CricaVPR [7] attend to salient features in the image without any explicit ROI extraction.

2.1.5 OTHER INTERESTING APPROACHES TO VPR

Other interesting approaches to place recognition include semantic-segmentation-based VPR [101], [102], [103], [104], [105], and object-proposals-based VPR [106], as reviewed by Garg *et al.* [107]. For images containing repetitive structures, [108] proposed a robust mechanism for collecting visual words into descriptors. Synthetic views are utilized for enhanced illumination-invariant VPR in [72], which shows that highly condition-variant images can still be matched, if they are from the same viewpoint. In addition to image retrieval, relevant research has been performed in semantic mapping to select images for insertion into a metric, topological, or topometric map as nodes/places. Semantic mapping techniques are usually annexed with VPR image retrieval techniques for real-world Visual-SLAM, please see the survey by [109]. Most of these semantic mapping techniques are based on Bayesian-surprise [110], [111], coresets [112], region proposals ([113]), change-point detection [114], [110], and salience-computation [115].

2.1.6 UNCERTAINTY ESTIMATION IN VPR

This topic has received less attention in VPR literature compared to viewpoint and appearance changes. Most works in VPR use the distance (e.g., L2 or Cosine) in feature space between a query and the nearest neighbor as the uncertainty estimate [9], or the distance between the retrieved nearest neighbors [116]. Some more recent works model the aleatoric uncertainty in image retrieval, e.g., the Bayesian Triplet Loss (BTL) [117] and the Self-Teaching Uncertainty Estimation (STUN) [118]. Both BTL and STUN estimate the aleatoric uncertainty in the training data by representing images as distributions instead of point estimates in the feature space. At training time, the loss function tries to bring the distributions of images belonging to the same place closer together in the feature space and pushes apart the distributions belonging to different places. Each image thus has an associated mean and variance for a feature descriptor.

Gronat *et al.* [119] treat VPR as a classification problem by training place-specific classifiers, one for each place, where each classifier naturally outputs a confidence estimate for the corresponding pose. Pion *et al.* [120] approximate the pose of the query image by aggregating the pose hypotheses from the top-retrieved nearest neighbors, weighing each hypothesis based on the distance in the feature space. The variance of the aggregated pose represents uncertainty over the pose space. Notably, this concept of pose uncertainty has been modeled in these existing works [38, 119, 120] and other related tasks such as classical Particle Filters [121], but, to the authors' best knowledge, the uncertainty estimates

derived based on the distribution of pose hypotheses have not been studied as a proxy for image-matching uncertainty.

Beyond global descriptors-based VPR, in local feature matching-based image retrieval the inlier count (aka. geometric verification) has been used as an estimate of confidence [62]. Zeisl *et al.* [122] perform 2D-to-3D local feature matching to estimate a distribution over the possible query poses. The work of [123] uses such an inlier count from local feature matching and combines it with the pose distribution of retrieved images to estimate the confidence of localization. Since local features can appear in similar geometric configurations (geometric burstiness) across unrelated images, [124] proposes to use the pose information to downweight such matches in the inlier count. However, retrieving images based on local feature descriptors is computationally expensive, whereas VPR instead only efficiently compares global image descriptors. Absolute Pose Regression (APR) directly regresses the absolute pose given a camera image, and has also considered pose uncertainty estimation. Some approaches to uncertainty-aware APR include CoordiNet [125], Bayesian PoseNet [126], and HydraNet [127]. Unlike VPR, APR approaches do not generalize to new environments.

2.1.7 BENCHMARKS AND EVALUATION METRICS IN VPR

Within the performance evaluation landscape, if we broaden our scope, it is evident that many researchers have evaluated visual-based localization at scale, which has led to a rapid development in this domain. From the computer vision perspective, the well-established visual-localisation benchmark[1] has been hosted for the past few years as workshops in top computer vision conferences. This benchmark was initially focused on 6-DoF pose estimates, but also included VPR (image-retrieval) benchmarking by combining with the Mapillary Street Level Sequences (MSLS) dataset ([128]) in ECCV 2020, although MSLS is mainly focused on sequences. The benchmarks have usually been organised as challenges (which have their own dedicated utility), where relevant evaluation papers also exist, e.g., the detailed works from [74] and [73]. Google also proposed the Landmarks dataset with focus on both place/instance-level recognition and retrieval: Google Landmark V1 dataset [62] and Google Landmark V2 dataset [129]. These benchmark datasets (and other similar datasets like Oxford Buildings, Paris Buildings etc.) and their associated evaluation metrics serve great value to the landmark recognition/retrieval problem, but focus on a particular category of datasets containing distinctive architectures, which may not be the primary focus of the robotics-centered VPR community requiring localisation-estimates throughout a continuous traversal that may be indoor, outdoor, natural and any/all others. In this context, VGBench [9] focuses purely on evaluating VPR for different types of outdoor settings.

With the extensive applications of VPR and, therefore, the correspondingly large number of relevant evaluation metrics, a higher-level breakdown can consist of two categories: direct and indirect evaluation metrics. Direct evaluation metrics are those metrics that directly measure the performance of a VPR system based on the images retrieved by the system from a given reference database for a set of query images. This direct evaluation of VPR systems is the scope of this thesis and is discussed at length in the following paragraph. On the other hand, indirect evaluation metrics for VPR are those metrics where VPR is only a part of the particular system's pipeline. In such cases, the evaluation metric is

[1] www.visuallocalization.net

measuring the performance of the complete pipeline, where indirectly a good-performing VPR module contributes to but is not the only determinant of achieving higher overall system performance. Some key examples of such indirect metrics within the Visual-SLAM paradigm are Absolute-Trajectory-Error (ATE) and Relative-Pose-Error (RPE), as presented in the RGB-D Visual-SLAM benchmark by [130]. Another commonly observed pipeline for 6-DoF camera-pose estimation with respect to a given scene is VPR followed by local feature matching, where the VPR module provides the initial coarse location estimate, which is then refined by local feature matching to yield 6-DoF camera pose. In such a case, the overall pipeline evaluation indirectly estimates VPR performance, as done by [73]. The ICCV 2025 CrocoDL challenge[2], where the author participated with his VPR method named *Dera*[3], is also an example of indirect evaluation of VPR.

Within direct performance evaluation, the most dominant VPR evaluation metric in robotics literature [5] has been Area-under-the-Precision-Recall curves (denoted usually as AUC-PR or simply AUC), which tries to summarize the Precision-Recall curves in a single quantified value. AUC-PR favors techniques that can retrieve the correct match as the top-ranked image, thus favoring applications that require highly precise localization estimates. The reasons for the more common use of PR-curves instead of Receiver Operating Characteristics curves (ROC-curves) in VPR are the imbalanced nature of the datasets and the usual lack of true negatives in datasets/evaluations. There is extensive VPR literature employing AUC-PR, for example, [131], [132], [133], [134], [99], and [135]. Other than AUC-PR, F1-score has also been used in VPR evaluations predominantly by the robotics-focused VPR community, for example, by [136], [137], [138], [139], and [140], to list a few. However, metrics like AUC-PR and F1-score quantify the performance of a VPR technique without considering the geometric distribution of true-positives within the trajectory. But since robotics is mostly concerned with achieving localization every few meters, [141] presents a new metric/analysis to compute the VPR performance, using the maximum distance traversed by a robot without achieving a true-positive/localization/loop-closure. Ferrarini *et al.* [142] presented a new metric Extended Precision (EP) for VPR evaluation that is based on Precision@100% Recall and Recall@100% Precision. In an author's previous work ([75]), he had presented PCU (Performance-per-Compute-Unit) as an evaluation metric for VPR, which combines place recognition precision with feature encoding time.

Recall@N (or RecallRate@N) is a dominant evaluation metric in the computer vision VPR community, which considers a retrieval to be true-positive for a given query if the correct ground-truth image is within the Top-N retrieved images. Recall@N has been used by e.g., [143], [108], [57], [72], [80] and [144]. For multiple correct matches in the database, Recall@N does not consider how many of the correct matches for a given query were retrieved by a VPR technique, therefore mean-Average-Precision (mAP) has also been extensively used by the computer vision VPR/image-retrieval community. Some of the literature that has employed mAP as an evaluation metric for VPR includes [145], [18], [124], [94], [96], and [129]. Other than these metrics, Recall@Reduced Precision has also been used as an evaluation metric ([146]) for place recognition. For computational analysis, feature encoding time, descriptor matching time, and descriptor size have been the key metrics for both communities.

[2] https://www.codabench.org/competitions/9471/
[3] https://github.com/MubarizZaffar/Dera

A large number of evaluation metrics can be employed for assessing the performance of a VPR system, and the selection is usually dependent upon the underlying application. However, it is also possible for the metrics from one community to be of value to the other community, such that the above-discussed distribution of metrics does not depict absoluteness but only dominant trends/applications. For example, Recall@N and Recall@Reduced Precision are also useful for robotic systems that can discard a small number of false-positives, e.g., by using outlier rejection in SLAM, false-positive prediction, ensemble-based approaches, and geometric verification. Similarly, mAP-based evaluations can support the creation of additional constraints for map optimization in SLAM.

2.2 OTHER TYPES OF VISUAL-BASED LOCALIZATION

The various other types of visual-based localization are comprehensively reviewed here in this section.

2.2.1 ABSOLUTE POSE ESTIMATION

Absolute Pose Estimation (APE) is the task of regressing a 6-DoF camera pose given an input query image. The first work in this domain was PoseNet [19], which trained a CNN to directly regress a camera pose. Hourglass-Net [147] uses an encoder-decoder architecture for direct pose regression given an input camera image. Instead of single image-only pose regression, temporal information is exploited in VidLoc [148] to account for perceptual-aliasing. Authors in [149] argue for the use of both relative and absolute pose constraints for training an absolute pose regressor. The authors in this case also advocate for the use of visual odometry at inference to provide relative pose constraints at inference time and implement the pose estimation within a pose graph optimization module. Since standard absolute pose estimation cannot generalize to new environments, ACE [20] proposes to divide the APE problem into two stages: feature extraction and pose regression. The former is learned as a generic map-agnostic feature extractor, whereas the latter is re-trained for new environments but in significantly lower time ($\approx$ 5 minutes in total) than existing approaches. Scene Coordinate Regression (SCR) can also be classified as a form of APE, since the objective is to learn absolute 3D scene coordinates for all pixels in a camera image [150–152]. MaRePo [153] follows a two-stage pose regression formulation, where a light-weight map-aware CNN learns scene coordinates while a map-agnostic heavier transformer learns to regress 6-DoF camera pose.

2.2.2 RELATIVE POSE ESTIMATION

Relative pose estimation is the task of estimating the relative camera pose between two input images. This could be further divided into two paradigms: regression-based approaches and correspondence-based approaches. Earlier works in regression used a CNN to regress relative pose between two images [21]. This work is followed up by a coarse-to-fine pose regression approach in [154]. Authors in [155] propose to use synthetic data for training a new relative pose regression architecture. Zhoue *et al.* [156] propose to learn to regress the essential matrix, which is then decomposed into a relative pose. Reloc3R [157] combines relative pose estimation with a motion averaging module to accurately estimate the relative camera pose. Khatib *et al.* [158] ground the relative pose regression on image matching using

LoFTR features to regress the relative camera pose. Relative pose can also be estimated using local feature matching, albeit with scale ambiguity. Examples of local feature descriptors that have been used for this task include SuperPoint [60], LoFTR [159], R2D2 [59], and MAST3R [160], etc. These local feature descriptors could be matched with different types of architecture, such as SuperGlue [161], LightGlue [162], Fast-NN matcher [160], etc. More recent task agnostic networks such as VGGT [163], Pow3R [164], and MapAnything [165] can also be used to regress relative camera poses given input images and, in some cases, camera intrinsics with optional extrinsics. Since standard SfM-based point clouds can be sparse, Giang *et al.* [166] present a method that infers semi-dense depth maps from sparse correspondences to achieve accurate localization without increasing computational costs.

2.2.3 STRUCTURE-BASED LOCALIZATION

Structure-based localization is the task of estimating the camera pose given an input query image and a 3D map. These methods achieve highly accurate localization by matching 2D-to-3D features, and then solving for camera pose using Perspective-n-point in a RANSAC loop or using Direct Linear Transform (DLT). Hloc [31] presents a modular structure-based localization pipeline, where first a coarse pose is estimated using image retrieval, followed by local feature matching and a PnP-RANSAC loop. Active-search is proposed by Sattler et al [167] where once a 2D-to-3D feature correspondence is achieved, the neighbourhood 3D features are then matched as 3D-to-2D feature matching, thus significantly reducing the search space. Zeis *et al.* [122] exploit known priors to design an O(n) complexity (in the number of matches) pose estimation method from a set of 2D-to-3D feature correspondences for reducing computational time. HyperPoints are proposed by Sattler *et al.* [168] where only locally unique points are considered for structure-based localization. InLoc [32] used dense feature matching instead of local feature matches for structure-based localization, followed by pose verification by virtual view synthesis to cope with significant changes in viewpoint.

2.2.4 CROSS-VIEW LOCALIZATION

Cross-view localization comprises two sub-tasks: cross-view retrieval and cross-view pose estimation. The former has generally been of less research interest and is marginalized by assuming access to a weak location prior. The latter has gained significant traction recently. Parallels can be drawn here from VPR and RPE, where the two tasks become similar to cross-view image retrieval and cross-view pose estimation, respectively, if the reference image is an aerial image and not a ground image. Similar to VPR, early work in cross-view localization also used handcrafted local features, such as SIFT [39] or image patches [169].

Trends changed with the rise of deep learning, with the early use of deep learning for cross-view image retrieval dating back to 2015 [170, 171]. Given the trend of deep representation learning using Siamese architectures, and its established use case in the VPR community, early works in cross-view image retrieval also used Siamese-style networks, albeit where the ground and aerial branches do not share weights [172, 173]. Because ground and aerial images represent two different domains, naturally, several works in cross-view image retrieval aim to bridge the domain gap by synthesizing one view from the other. Synthetic aerial images are created from ground images using GANs in [174]. SAFA [173] uses a polar transformation on the aerial image to synthesize an image visually similar to

the ground image. Transformers have also been used in cross-view image retrieval, thus benefiting from their strong attention mechanism [175, 176].

Once a coarse location estimate is available, either through cross-view image retrieval or a weak location prior, the task becomes localizing the query image in its corresponding aerial image. Cross-view pose estimation was originally formulated as a regression problem [23, 177]. However, it can also be approached as feature matching problem, e.g., Vision Transformers have been used to map the features of the ground-level images to Bird's Eye View (BEV), which are then densely compared to feature maps extracted from the aerial image [178]. SNAP [179] fused information from ground-level and aerial images to construct a so-called neural map, which is then used for localizing ground-level images. Xia *et al.* propose Convolutional Cross-View Pose Estimation, where ground and aerial images are encoded as orientation-aware image descriptors and convolved together to retrieve accurate location and orientation [180].

2.3 TAKE-AWAYS FROM THE REVIEWED LITERATURE

The literature reviewed in this thesis first showcases that VPR indeed is a challenging and important research task for both the robotics and computer vision communities, and has applications for other types of VBL too. It has been approached from both the standard handcrafted and recent deep-learning approaches, where the former tends to be much less accurate but efficient than the latter. The rise of vision foundation models and stronger feature extraction backbones has directly benefited the VPR community. There are two tangential tracks, aiming for the universally best VPR method and designing domain adaptation techniques for VPR. The hypothesis for foundation-models-based VPR methods being universally applicable or if they could benefit from domain adaptation is untested. While significant attention has been paid to improving the robustness of VPR methods, there is limited literature on the uncertainty estimation in VPR. It is observed that most works assume the feature-space distance as the reliable uncertainty estimation metric in VPR, but this remains untested. There is also a disconnect between the evaluations performed and considered relevant for the robotics and computer vision communities. A state-of-the-art on one metric may not be the best on another. The choice of popular evaluation metric also seems to be dominated by the image retrieval community, where it remains unclear how improved performance could benefit applications such as localization and SLAM. VPR has primarily been studied stand-alone, and its formulation as a localization task or a reliable loop-closure proposer is not a dominant trend in the literature.

3

VPR-Bench: The First Open-source Benchmark in Visual Place Recognition

This chapter is based on *M. Zaffar. VPR-Bench: An Open-Source Visual Place Recognition Evaluation Framework with Quantifiable Viewpoint and Appearance Change, IJCV, 2021.* [8].

Author contributions: Mubariz Zaffar proposed and implemented the framework, performed the experiments, and took the lead in writing and presenting. Sourav Garg assisted with running some additional experiments and helped with the writing. All other authors provided feedback on the technical writing and visualizations.

3.1 OVERVIEW

The previous chapters introduced and reviewed Visual Place Recognition (VPR): a challenging and widely investigated problem within the computer vision community ([5]). VPR identifies the ability of a system to match a previously visited place, with resilience to perceptual aliasing and seasonal-, illumination- and viewpoint-variations. This ability to correctly and efficiently recall previously seen places using only visual input has many important applications, such as loop-closure in SLAM (Simultaneous Localisation and Mapping) pipelines ([25]) to correct for localization drifts, image search based on visual content ([181]), location-refinement given human-machine interfaces ([182]), query-expansion ([183]), improved representations ([184]), vehicular navigation ([185]), asset-management using aerial imagery ([186]) and 3D-model creation ([26]).

Consequently, VPR researchers come from various backgrounds, as witnessed by the many workshops organised in top-tier conferences, e.g. 'Long-Term Visual Localisation Workshop Series' in Computer Vision and Pattern Recognition Conference (CVPR), 'Visual Place Recognition in Changing Environments Workshop Series' in IEEE International Conference on Robotics and Automation (ICRA), 'Large-Scale Visual Place Recognition and Image-Based Localization Workshop' in IEEE International Conference on Computer Vision (ICCV 2019) and 'Visual Localisation: Features-based vs Learning Approaches' in European Conference on Computer Vision (ECCV 2018). Thus, VPR has drawn huge interest from the computer vision and robotics research communities, leading to a large number of VPR techniques proposed over the past many years, but the communities remain separated and the state-of-the-art is not temporally consistent (see Fig. 3.2).

This divide is primarily due to the application requirements for both the domains: robotics researchers usually focus on having highly confident estimates predicting a revisited place to perform loop-closure, while the computer vision community prefers to retrieve as many prospective matches of a query image as possible for 3D-model creation, for example. The number of correct reference matches for the former are usually limited to a few (1-5), associated with repeated traversals under varied conditions, and thus robotics uses smaller datasets, e.g. Gardens Point dataset ([187]), ESSEX3IN1 ([115]) dataset, Campus Loop dataset ([82]) and others. For the latter, the number of correct matches (reference images) are larger (> 10), corresponding to a broad collection of photos of a landmark, and thus uses substantially sized datasets, e.g. the Pittsburgh dataset ([108]), Oxford Buildings dataset ([188]), Paris dataset ([189]) and their revisited versions with increased 1M distractors by [190].[1] In addition, robotics mostly focuses on high precision, usually requiring a single correct match for localisation estimates. It therefore employs evaluation metrics such as AUC-PR and F1-Score, while the computer vision community has predominantly used Recall@N, mean-Average Precision (mAP) and/or Recall@Reduced Precision. The divergence in datasets and metrics has limited the comparison of the techniques across the two domains to intra-domain-type evaluations, hence the state-of-the-art remains ambiguous. Therefore, one of the key contributions of this chapter is attempting to reduce this gap by integrating datasets, metrics and techniques from both the domains into a novel framework

[1] These remarks are only depicting the evident trends and are not absolute. Large-scale datasets (e.g. the Nordland dataset by [191] and Oxford robot-car dataset by [192]) for the robotics community, and small-scale datasets (e.g. the INRIA Holidays dataset by [145]) for the computer vision community do exist.

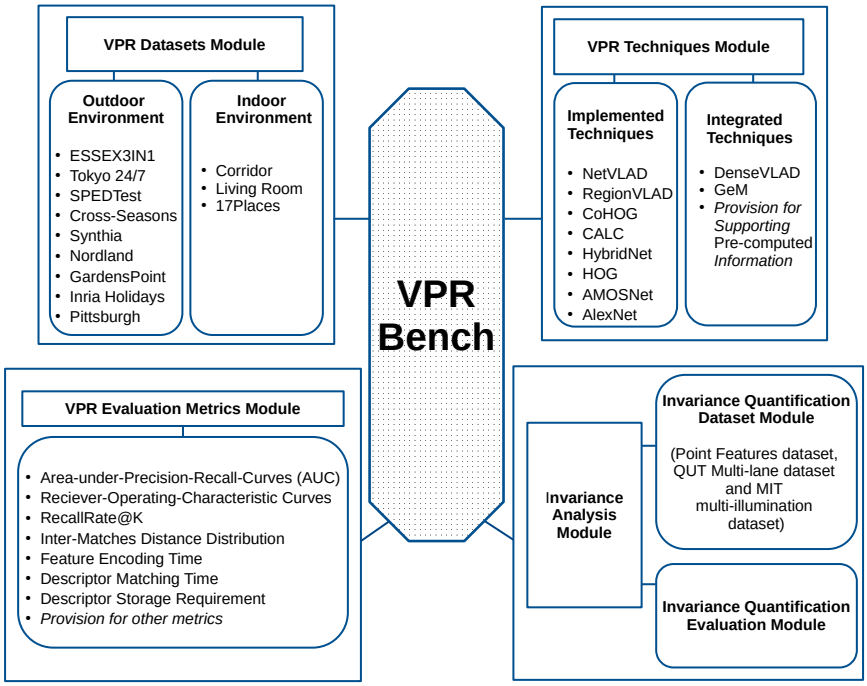


Figure 3.1: A block-diagram overview of the developed VPR-Bench framework is shown here. All modules can be inter-linked within the framework and can also be independently modified for graceful updates in the future.

called *VPR-Bench*, which is carefully designed to add convenience and value for both communities.

Moreover, a significant body of VPR research has focused on proposing techniques that are invariant to viewpoint, illumination and seasonal variations, all of which are major challenges in VPR. However, these techniques have usually been assessed qualitatively in the past using a rough categorisation of invariance such as 'mild', 'moderate', 'high' and 'extreme', etc., which are subjective and ambiguous. Although seasonal variations are difficult to quantify, viewpoint and illumination variations can be modelled by quantitative metrics. Therefore, another key focus of this research is to quantify the invariance of VPR techniques to viewpoint and illumination changes. The author utilises the detailed variation-quantified Point Feature dataset ([193]) and integrates it into the proposed framework to numerically and visually interpret the invariance of techniques. This quantified variation is obtained by taking images of a fixed scene from various angles and distances, under different illumination conditions, as explained later in sub-section 3.2.5. Since the Point Features dataset is a synthetically-created dataset, this chapters also include the QUT multi-lane dataset ([194]) and MIT multi-illumination dataset ([195]), which each respectively represent quantified variations in viewpoint and illumination in a real-world setting.

Furthermore, this chapter presents a detailed meta-analysis enabled by VPR-Bench. The author has integrated Receiver-Operating-Characteristic (ROC) curves into VPR-Bench to

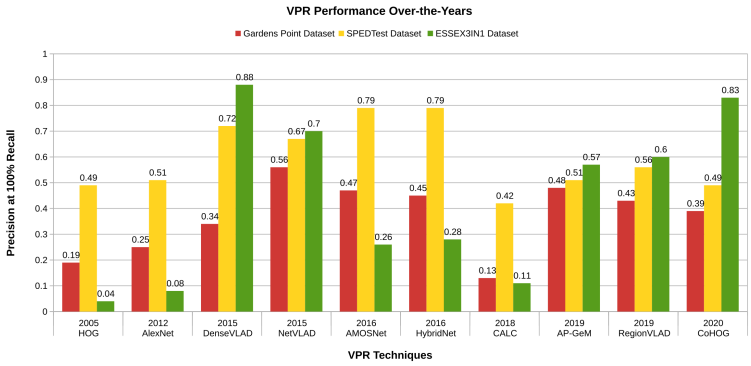


Figure 3.2: Precision at 100% Recall (equivalent to RecallRate@1) of 10 VPR techniques on Gardens Point dataset ([76]), SPEDTest dataset ([98]) and ESSEX3IN1 dataset ([115]) is shown here in a chronological order. The trends show irregularities in between techniques and datasets, while the increase in precision is also not temporally consistent. These datasets and techniques have been discussed later in this chapter. Please note that this graph is not intended to reflect the utility of these techniques, as some less-precise techniques have significantly lower computational requirements and can process more place-recognition (loop-closure) candidates.

analyse the ability of VPR techniques to find 'new places', i.e. true-negatives, which are generally not available in Precision-Recall type metrics. The chapter presents analysis on the distribution of true-positives within a sequence, which helps to understand the utility of VPR techniques based on spatial gaps between consecutive true-positives. In addition to the metric-based performance evaluations, this chapter also discusses case-studies on ground-truth manipulation that can lead to varying state-of-the-art, and the CPU vs GPU performance differences for deep-learning-based VPR techniques. The descriptor size of VPR techniques also affects VPR performance, and the author analyzes these effects in this chapter. The retrieval time of VPR techniques is compared with platform dynamics to yield insights into the relation between map-size, encoding-times, matching times, and platform velocity. A sub-section is dedicated to discussing the impacts and usage of viewpoint variance instead of invariance for VPR techniques in changing application scenarios. Finally, the source code for this comprehensive framework is made fully public, and all datasets with their associated ground-truths are re-released. An overview of the proposed framework is shown in Fig. 3.1.

In summary, the main contributions of this chapter are:

1. This chapter presents a systematic analysis of VPR by employing the largest collection of techniques, datasets and evaluation metrics to date from the computer vision and the robotics VPR communities, such that it accommodate a large number of scenarios, including very-small scale datasets to large-scale datasets, indoor to outdoor and natural environments, moderate to extreme viewpoint and conditional variations and several evaluation metrics that complement each other.

2. The author develops an open-source, fully-integrated, extensive framework for evaluating VPR performance. He re-implements a number of VPR techniques based on the proposed unified templates and restructures datasets and their ground-truths

into consistent and compatible formats, which are re-released, thus providing a pre-established go-to strategy for employing a variety of metrics, datasets, and popular VPR techniques for all new evaluations on a common ground.

3. This chapter quantifies the notion of viewpoint and illumination invariance of VPR techniques by employing a detailed variation-quantified Point Features dataset. The findings are further extended to 2 real-world, variation-quantified datasets, namely the QUT multi-lane dataset and the MIT multi-illumination dataset.

4. The author presents a number of different analyses within the VPR performance evaluation landscape, including the effects of acceptable ground-truth manipulation on rankings, the trade-offs between viewpoint variance vs invariance, the effects of descriptor size on the performance of a technique, the CPU vs GPU computational performance rankings and the trends of image retrieval times' variation with changing map-size on par with a platform's dynamic

3.2 METHODOLOGY

This section introduces the details of the proposed novel VPR-Bench framework, including the task formulation, datasets, techniques, evaluation metrics, and the invariance quantification module, respectively.

3.2.1 VPR TASK FORMULATION

The goal of VPR is to find one or multiple reference images $I_i \in \mathcal{I}_\mathcal{R}$ that match the place of a query image $I_q \in \mathcal{I}_Q$ given a set of reference images $\mathcal{I}_\mathcal{R}$ with known poses $\mathcal{P}_\mathcal{R}$. Usually, a VPR method G is applied to every reference image $I_i \in \mathcal{I}_\mathcal{R}$ to obtain D-dimensional reference feature descriptors $f_i = G(I_i)$. In some cases, the feature descriptor f_i is also to hold within it additional required information, e.g., the locations of regions-of-interest, their descriptors, and corresponding salience. This method G is mostly a trained neural network [196] or a handcrafted feature descriptor [197]. The resulting VPR map $\mathcal{M} = (\mathcal{I}_\mathcal{R}, \mathcal{R}, \mathcal{P}_\mathcal{R})$ contains the reference feature descriptors set $\mathcal{R} = \{f_1, \cdots f_N\}$, where each descriptor f_i is associated with a corresponding pose $p_i \in \mathcal{P}_\mathcal{R}$. At test-time, the method G is applied to the query image I_q, and its descriptor $f_q = G(I_q)$ is compared to the reference descriptors in the map $\mathcal{M}$. This can be achieved through an efficient K-nearest neighbor lookup, considering the L2-distances $d_i = \|f_i - f_q\|_2$ between each reference i and the query q. The score d_i generally ranges between 0-1, and can represent other distances, such as the cosine-distance, etc.

3.2.2 EVALUATION DATASETS

In this section, the existing patterns and features of datasets in VPR are presented, and each of the datasets that have been used in this chapter is discussed by dividing into outdoor and indoor datasets categories.

DATASET CONSIDERATIONS IN VPR-BENCH

All the datasets that have been employed to date for VPR evaluation comprise multiple views of the same environment that may have been extracted under different seasonal, viewpoint, and/or illumination conditions. These views are mostly available in the form of monocular

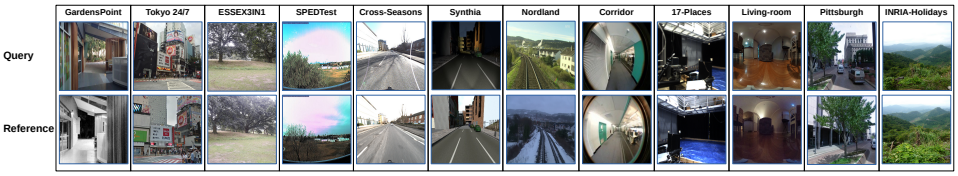


Figure 3.3: Sample images from all 12 VPR datasets employed in this chapter are presented here. These datasets span many different environments, including cities, natural scenery, train-lines, rooms, offices, corridors, buildings, busy-streets and such.

images and are structured as separate folders representing query and reference images. However, these views may have been extracted from a traversal or a non-traversal-based mechanism. For the former, consecutive images within a folder (query/reference) usually have overlapping visual content, while for the latter, images within a folder are independent. Accompanying these folders is usually some level of ground-truth information, which has been represented in various ways (e.g, CSV, numpy arrays, pickle files containing frame-level correspondence, GPS, pose information etc.) for different datasets. In some cases, the ground-truth is not explicitly provided, as images with the same index/name represent the same place.

For most traversal-based datasets, there is no single correct match for a query image, because images which are geographically close-by can be considered as the same place, leading to a range requirement for ground-truth matches instead of a single match/value. For such datasets and viewpoint-invariance in general, defining a correct ground-truth is 'tricky' because depending upon the acceptable level of viewpoint invariance for a VPR technique, the underlying ground-truth can be manipulated to change the performance ranking, as shown later in section 3.3.6. Another key challenge is the relation between visual-overlap, scene-depth, and physical distance. In an outdoor environment (e.g., highway), frames that are 5 meters apart may have significant visual overlap due to high scene depth, while frames that are 5 meters apart in an indoor environment may be visually very different due to low scene-depth and therefore frame-range-based ground-truth for most VPR datasets includes manual adjustment of ground-truth frame-range given visual overlap sanity checks.

Generally, there is a trade-off between pose accuracy and viewpoint invariance, where none of these can explicitly define a hard requirement from a VPR system. If a VPR system is being used as the primary localisation system (robotics perspective), higher pose accuracy is desired and the system should have *viewpoint-variance*, while for retrieving maximum matches of a place from the reference database (computer vision perspective), *viewpoint-invariance* is the key requirement. For the robotics perspective, pose inaccuracy can be reduced at increased computational cost by using relative pose estimation as a subsequent pose refinement stage. Therefore, some viewpoint invariance (usually defined by a few meters) has always been required from a VPR system in both the communities. To address this 'loose' nature of viewpoint-invariance definition of a VPR system, the author has taken the following steps:

1. The author has integrated datasets that contain a large variation in the acceptable ground-truth viewpoint variance: ranging from the minimally acceptable viewpoint

Table 3.1: The 12 VPR-Bench datasets integrated into VPR-Bench and used in this study are enlisted here. The sign ˜ for image resolutions (pixels × pixels) indicates datasets where image resolution varies in-between different images of the dataset, and therefore the common resolution observed in that dataset is specified.

Dataset	Environment	Queries	References	Viewpoint Change	Conditional Change	Query Res.	Ref Res.
GardensPoint	University Campus	200	200	Lateral	Day-Night	960 × 540	640 × 360
Tokyo 24/7	Outdoor	315	75984	3D	Day-Night	3264 × 2448	640 × 480
ESSEX3IN1	University Campus	210	210	3D	Illumination	720 × 720	1080 × 1080
SPEDTest	Outdoor	607	607	None	Seasonal and Weather	320̃ × 240̃	320̃ × 240̃
Cross-Seasons	City-like	191	191	Lateral (Occasional)	Dawn-Dusk	1024 × 1024	1024 × 1024
Synthia	City-like (Synthetic)	813	911	Lateral	Time and Season	300 × 200	300 × 200
Nordland	Train Journey	2760	27592	None	Seasonal	640 × 360	640 × 360
Corridor	Indoor	111	111	Lateral	None	160 × 120	160 × 120
17-Places	Indoor	406	406	Lateral	Day-Night	640 × 480	640 × 480
Living-room	Indoor	32	32	Lateral	Day-Night	1792 × 896	1792 × 896
Pittsburgh	Outdoor	1000	23000	3D	None	640 × 480	640 × 480
INRIA Holidays	Outdoor	300	512	Lateral/3D	None	250̃ × 185̃	250̃ × 185̃

variation in the Corridor dataset to the large acceptable viewpoint variations of the Tokyo 24/7 dataset, thus to cover a broader audience.

2. An extensive analysis on the effects of changing acceptable levels of viewpoint invariance is provided in section 3.3.6.

3. As for consistency in VPR research and performance reporting, it is essential to affix a unified template for all of these VPR datasets. Thus, the author has re-released all datasets in a VPR-Bench compatible format with their associated ground-truth information.

Despite the extensive collection of datasets in this chapter, there are still scenarios which are not represented in these datasets, e.g., extreme weather conditions, aerial and underwater platforms, opposing views, and motion-blur resulting from high-speed platforms. The author has designed VPR-Bench as per unified templates to allow integration of new datasets.

OUTDOOR ENVIRONMENT

The author has integrated multiple outdoor datasets in the proposed framework representing different types and levels of viewpoint-, illumination- and seasonal-variations. Details of these datasets have been summarised in Table 1, and sample images are shown in Fig. 3.3. Each of these datasets has a particular attribute to offer, that leads to its selection, and they are briefly discussed below.

The GardensPoint dataset was created by [187] and first used for VPR by [76], where two repeated traversals of the Gardens Point Campus of Queensland University of Technology, Brisbane, Australia were performed with varying viewpoints in day and night times. A huge body of robotics-focused VPR research has used this dataset for reporting their VPR matching performance, as it depicts outdoor, indoor, and natural environments, collectively. The author has only used the day and night sequences in this chapter because they contain both the viewpoint and conditional change. The Tokyo 24/7 dataset was proposed by [72], which consists of 3D viewpoint-variations and time-of-day variations. The author uses version 2 of the query images, as suggested by Tori et al. [72] and Relja et al. [80] to maintain comparability. It is one of the most challenging datasets for VPR due to the sheer amount of viewpoint- and conditional-variation, and has been used by both the robotics and

vision communities. The ESSEX3IN1 dataset was proposed by Zaffar et al. [115] and is the only dataset designed with a focus on perceptual aliasing and confusing places/frames for VPR techniques. The SPEDTest dataset was introduced by Chen et al. [98] and consists of low-quality, high scene-depth frames extracted from CCTV cameras across the world. This dataset has the unique attribute of covering a huge variety of scenes from all across the world under many different weather, seasonal and illumination conditions. The Synthia dataset was introduced in [198] and represents a simulated city-like environment in various weather, seasonal and time of day conditions. In this chapter, the author has used the night images from Synthia Video Sequence 4 (old European town) as query and the fog images as reference from the same sequence. The Cross-Seasons dataset employed in this chapter represents a traversal from [199], which is a subset of the Oxford RobotCar dataset ([192]). This dataset represents a challenging real-world car traversal from dawn and dusk conditions. One of the widely employed datasets for VPR is the Nordland dataset, developed by [191] and introduced to VPR evaluation by [200], which represents a 728 kilometers of train journey in Norway during the Summer and Winter seasons. As Nordland dataset represents a natural (non-urban), outdoor environment, which is unexplored in any other dataset, it is integrated it into VPR-Bench. From the computer vision community, in addition to Tokyo 24/7, the author uses the Pittsburgh dataset ([108]) and the INRIA Holidays dataset ([145]) to bridge the important gap between the two communities. The author only uses the query images of Pittsburgh dataset because this represents the only large-scale dataset in the proposed framework that has 3D viewpoint-variation without any conditional variation. The INRIA Holidays dataset, similar to the SPEDTest dataset, explores a very large variety of scenes but also includes indoor scenes as well, and uses the highly relevant egocentric viewpoint unlike the CCTV-based SPEDTest. These datasets are still only a subset from an apparent zoo of datasets available for VPR evaluation. Despite the large number of outdoor datasets used in this work, there are still scenarios that are not covered here, including extreme weather conditions, opposing views, motion-blur, aerial, and underwater datasets.

INDOOR ENVIRONMENT

A significant focus in recent research in VPR has primarily been on evaluation on outdoor datasets, so indoor environments are also incorporated into VPR-Bench, which are usually a key area of study within robot autonomy. While indoor datasets, usually do not represent the seasonal variation challenges as outdoor datasets and the level of viewpoint-variation is relatively lesser than outdoor datasets, they do contain dynamic objects like humans, animals, or changing setup/environment configurations, less-informative content, and perceptual-aliasing. The details of these datasets have been summarised in Table 1, and sample images are shown in Fig. 3.3. The currently available indoor datasets in VPR-Bench have been briefly discussed, in the following paragraph.

The author has integrated the 17-Places dataset introduced by [201] into VPR-Bench, which consists of a number of different indoor scenes, ranging from office environment to labs, hallways, seminar rooms, bedrooms and many other. This dataset exhibits both viewpoint- and conditional-variations. The author also uses the viewpoint-variant Corridor dataset, introduced by [202], which represents the challenge of low-resolution and feature-less images (160×120 pixels) for vision-based place recognition. [203] introduced the living-room dataset for home-service robots, which represents indoor environment from a highly relevant and challenging viewpoint of cameras mounted close-to-ground level.

Table 3.2: The ground-truth tolerance for the 12 VPR-Bench datasets integrated into VPR-Bench is provided here. The † next to Pittsburgh dataset indicates that 23 ground-truth images are available for every query image, taken at different pitch and yaw angles without any translational movement of the camera.

Dataset	Ground-truth Tolerance
GardensPoint	± 2 frames
Tokyo 24/7	± 25 meters
ESSEX3IN1	Frame-to-frame
SPEDTest	Frame-to-frame
Cross-Seasons	± 5 meters
Synthia	± 7 meters
Nordland	± 1 frames
Corridor	± 2 frames
17-Places	± 3 frames
Living-room	± 2 frames
Pittsburgh	23 frames †
INRIA Holidays	Frame-to-frame

This dataset only contain 32 queries and 32 references, the author deliberately uses such a small-scale dataset to see the ordering of VPR techniques on very small-scale datasets.

GROUND-TRUTH INFORMATION

Because the author has utilised a variety of different datasets from both the robotics and the computer vision communities, which are also from both indoor and outdoor environments, the underlying ground-truth information is varying. The author has thus used the ground-truth information provided by the original contributors of these datasets (or in some cases the modified ground-truths used in recent evaluations) and reformatted these into ground-truth compatible to the templates developed for VPR-Bench. All the datasets and their ground-truths are re-released, and therefore this ground-truth information is only briefly presented in Table 3.2. The ground-truth tolerance for some of the robotics-focused VPR datasets is strict in comparison to the computer vision datasets when it comes to viewpoint variance/invariance, i.e., the reference images that are geographically far apart but have some visual overlap are not considered as correct matches for the robotics datasets. Instead of relaxing the viewpoint variance for the robotics datasets and/or restricting the viewpoint variance for the computer vision datasets, the original levels being used by their respective communities are retained.

3.2.3 VPR TECHNIQUES

In this section, the author introduces the 10 VPR techniques that have been evaluated in this chapter, while also providing important implementation details of these techniques that are needed to understand the experiments and results in the next section 3.3.

HOG-Descriptor: Histogram-of-oriented-gradients (HOG) is one of the most widely used handcrafted feature descriptors, which actually performs very well for VPR compared to other handcrafted feature descriptors. It is a good choice for a traditional handcrafted

feature descriptor in the proposed framework, based upon its performance as shown by [71] and the value it presents as an underlying feature descriptor for training a convolutional auto-encoder in [82]. The author uses a cell size of 16×16 and a block size of 32×32 for an image-size of 512×512. The total number of histogram bins is set equal to 9. The author uses cosine-matching between HOG-descriptors of various images to find the best match.

AlexNet: The use of AlexNet for VPR was studied by [137], who suggest that *conv3* is the most robust to conditional variations. Gaussian random projections are used to encode the activation-maps from *conv3* into feature descriptors, and cosine distance is used for matching. The author's implementation of AlexNet is similar to the one employed by [82], while the code has been restructured as per the designed template. Note that AlexNet resizes the input image to 227×227 before it is input to the neural network.

DenseVLAD: DenseVLAD has been proposed by [72], where they densely-sample local SIFT keypoints from images, corresponding to regional widths. These keypoints are extracted at 4 different scales. The local keypoints are then converted into a global descriptor using a Vector-of-Locally-Aggregated-Descriptors (VLAD) dictionary consisting of 128 visual-words extracted by K-means clustering on a dictionary of 25M randomly-sampled descriptors. PCA-compression and whitening is performed on the final descriptor to down-sample it into a 4096-dimensional descriptor. The author has formatted (as per the proposed template) and integrated the descriptor matching data computed by the DenseVLAD code open-sourced by [72] into VPR-Bench to demonstrate the utility of the proposed framework for cases where code conversion may not be required/desired. All input images are resized to 640×480, similar to [72].

AP-GeM: GeM was originally proposed by [85], where they presented a new generalised-mean layer to replace the typical max-pooling and sum-pooling for feature descriptor mining from a CNN tensor. This was then upgraded by [96], where they have designed a new ranking-loss based on mean-Average-Precision. The author has used the GeM code open-sourced by [96] based on the ResNet101 model (namely ResNet101-AP-GeM) with an output descriptor size of 2048 dimensions. Similar to DenseVLAD, the author has used the descriptor matching data computed by the original code of the respective authors and integrated that with the proposed framework for a seamlessly straightforward integration process. [96] used 800×800 resolution for training but performed no resizing during testing. Thus, for a fair comparison against other input resolution-independent methods such as NetVLAD and DenseVLAD, the author resized input images to 640×480.

NetVLAD: The original implementation of NetVLAD was in MATLAB, as released by [80]. The Python port of this code was open-sourced by [204]. The model selected for evaluation is VGG-16, which has been trained in an end-to-end manner on Pittsburgh 30K dataset ([80]) with a dictionary size of 64 while performing whitening on the final descriptors. The code has been modified as per VPR-Bench template. The authors of NetVLAD have suggested an image resolution of 640×480 at inference time, and therefore, this image resolution is used for all experiments.

AMOSNet: This technique was proposed by [78], where a CNN has been trained from scratch on the SPED dataset. The authors have presented results from different convolutional layers by implementing spatial-pyramidal pooling on the respective layers. While the original implementation is not fully open-sourced, the trained model weights have been shared by the authors. AMOSNet is implemented as per VPR-Bench template using *conv5* of the shared model. L1-match has been originally proposed by the authors, which is normalised for a score between $0-1$. The default AMOSNet resizes input images to 227×227.

HybridNet: While AMOSNet was trained from scratch, [78] took inspiration from transfer learning for HybridNet and re-trained the weights initialised from Top-5 convolutional layers of CaffeNet ([79]) on SPED dataset. HybridNet is implemented as per the template using *conv5* of the HybridNet model. L1-match has been originally proposed by the authors, which is normalised for a score between $0-1$. The default implementation of HybridNet resizes input images to 227×227.

RegionVLAD: Region-VLAD has been introduced and open-sourced by Khaliq et al. [99]. The author has modified it as per the proposed template and has used AlexNet trained as Places365 dataset as the underlying CNN. The total number of ROIs has been set to 400, and 'conv3' is used for feature extraction. The dictionary size is set to 256 visual words for VLAD retrieval. Cosine similarity is subsequently used for matching descriptors of query and reference images. The default RegionVLAD resizes input images to 227×227.

CALC: The use of convolutional auto-encoders for VPR was proposed by [82], where an auto-encoder network was trained in a weakly-supervised manner to re-create similar HOG-descriptors for viewpoint-variant (cropped) images of the same place. The author uses model parameters from $100,000$ training iterations and adapts the open-source technique as per the template. Cosine-matching is used for descriptor comparison. This is the only semi-supervised learning technique in proposed framework and therefore has its own particular utility. The default implementation of CALC resizes input images to 120×160.

CoHOG: CoHOG is a recently proposed ([75]) handcrafted feature-descriptor-based technique, which uses image-entropy for ROI extraction. The regions are subsequently described by dedicated HOG-descriptors, and these regional descriptors are convolutionally matched to achieve lateral viewpoint-invariance. It is an open-source technique, which has been modified as per VPR-Bench template. An image-size of 512×512, cell-size of 16×16, bin-size of 8, and an entropy-threshold (ET) of 0.4, is used. CoHOG also uses cosine-matching for descriptor comparison.

3.2.4 EVALUATION METRICS

A trend within current VPR research has shown that a single, universal metric to evaluate VPR techniques that could simultaneously extend to all applications, platforms and user-requirements does not exist. For example, a technique which has a very high-precision, but a significantly higher image-retrieval time (few seconds), may not extend to a VPR-based, real-time topological navigation system, as the localisation module will be much slower (in

Table 3.3: A taxonomy of VPR evaluation metrics is given here. Where PL: Primary Localisation, LC: Loop-closure, IR: Image Retrieval, FP: False-Positives, RC: Robotics Community, CV: Computer Vision Community, MB: Matching-based and CB: Computational-intensity-based. * identifies a sub-class of PL and LC, where the underlying system is not robust to false-positives,. This robustness normally arises from geometric-verification, visual-inertial odometry, re-ranking schemes, false-positive predictors, weak-prior and/or other similar modules.

Metric	Primary Usage	Output	FP Allowed?	Primary Audience	Nature
AUC-PR	PL+LC+IR	Single-value	Yes	RC+CV	MB
Extended Precision	PL*+LC*	Single-value	No	RC	MB
Recall@100%Precision	PL*+LC*	Single-value	No	RC	MB
RecallRate@N	PL+LC+IR	N-values	Yes	RC+CV	MB
Recall@ReducedPrecision	PL+LC+IR	Single-value	Yes	RC+CV	MB
mean-Average-Precision	IR	Single-value	Yes	CV	MB
F1-Score	PL+LC	Multiple-values	Yes	RC+CV	MB
Encoding Time	PL+LC	Single-value	Yes	RC	CB
Matching Time	PL+LC+IR	Single-value	Yes	RC+CV	CB
PCU	PL+LC	Single-value	Yes	RC	MB+CB
RMF	PL+LC	Single/Multiple values	Yes	RC	MB+CB

frames-per-second processed) than the platform dynamics. However, for situations where real-time place matching may not be required, for example, offline loop-closures for map correction, improved representations and structure-from-motion, high precision at the cost of higher retrieval time may be acceptable. Therefore, reporting performance on a single metric may not fully present the utility of a VPR technique to the entire academic, industrial, and research audience, and the application-specific communities within them. The author has integrated into VPR-Bench, a variety of different metrics that evaluate a VPR technique on the fronts of matching performance, computational needs, and storage requirements.

The taxonomy of various metrics used in VPR by both the computer vision and the robotics communities is collated in Table 3.3 for the reader's reference, which are also discussed later in the chapter. The primary usage and audience of the techniques do not represent the limitations of the respective metrics to particular use-cases/communities, but instead identify the best/most-suitable use-cases for the respective metric. The usage is broadly classified into 3 areas: primary-localisation, loop-closure, and image-retrieval. Each of these classes can then contain various applications, e.g., image-retrieval (which intends to retrieve as many correct matches for a query as possible from the database) could be used for query-expansion, structure-from-motion (3D-model creation), content-based search engines, and many others. Primary-localisation (a vision-only localisation system that uses VPR for position estimates) and loop-closure (error drift correction in a SLAM pipeline) do not require the retrieval of all the existing matches of a query from the database, but instead a single (or few) correct match(es) to have a location estimate at a high frame-rate. A primary-localisation system may or may not have a false-positive rejection scheme within its localisation pipeline, and therefore, the respective application and the suited metric would change accordingly. Loop-closure represents an important VPR application within a visual-SLAM system. Because the objective of having loop-closure is to correct the existing uncertainty of the visual-SLAM system, it is usually preferred that a highly precise VPR technique be used for loop-closure. The *kidnapped robot problem* can also be considered as a particular case of loop-closure. In the following, the author discusses each of the metrics that have been used for evaluations in this chapter, their motivation, and limitations.

AUC AND PR-CURVES

Motivation: AUC-PR is one of the most used evaluation metrics in the robotics VPR community. It presents a good overview of the precision and recall performance of a VPR technique, where only a single correct match, which should be the best matched reference image, is required for a given query image. Therefore, it is usually suitable for applications that require high precision, high recall, a single correct match, and that only consider the best matched image for their operation, e.g., loop-closure and topological-localisation.

Limitations: AUC-PR may not be relevant for applications that intend to retrieve as many correct ground-truth matches as possible from the reference database. It is not affected if the second-best (or third-best, and so on) match is actually a correctly retrieved image. Thus, it has two major limitations: in cases where many correct ground-truth matches exist in the database and the system application (3D-modelling, constraint-creation) requires the correct retrieval of all of these images, AUC-PR may not present significant relevance, as it only considers a single retrieved image per query in its computations. Secondly, AUC-PR may not be relevant in cases where false-positive rejection is possible (e.g., weak GPS prior, geometric verification, robust optimization back-ends) and the VPR system is mainly used to retrieve a correct match within a list of top matching candidates.

Metric design: AUC-PR is computed from Precision-Recall curves, which are aimed at understanding the loss of precision with increasing recall at different confidence score thresholds. Generally, in VPR the image similarity scores are considered as confidence scores and are varied within the maximum range to plot PR-curves. Precision and Recall are computed for each threshold in a range of thresholds as

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}, \tag{3.1}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}. \tag{3.2}$$

Where in terms of VPR, given a query image and a chosen confidence score threshold, a True-Positive (TP) represents a correctly retrieved image of a place based on ground-truth information. A False-Positive (FP) represents an incorrectly retrieved image based on ground-truth information. A False-Negative (FN) is a correctly retrieved image based on ground-truth, the matching score for which is lower than the chosen confidence score threshold. Please note that in most VPR datasets, all correctly matched images that are rejected due to the matching scores being lower than the chosen threshold are classified as false-negatives, because ground-truth matches exist for all images in the datasets. There are *no* True-Negatives (TN) usually in the datasets, i.e., query images that do not have a correct match in the reference database (the author also discusses this later in the chapter for ROC curves). By selecting different values of the matching threshold, varying between the highest matching score and the lowest matching score, different values of Precision and Recall can be computed. The Precision values are plotted against the Recall, and the area under this curve is computed, which is termed AUC-PR. The ideal value of AUC-PR is 1, and Precision=1 for all recall values represents an ideal PR-curve.

RECALLRATE@N

Motivation: One of the most commonly used evaluation metrics from the computer vision VPR community is RecallRate@N (also termed as Recall@N). This metric tries to model the

fact that a correctly retrieved reference image (as per the ground-truth) does not necessarily have to be the top-most retrieved image, but only needs to be among the Top-N retrieved images. The primary motivation behind this is that subsequent filtering steps, e.g., geometric verification or weak GPS-prior, can be used to re-arrange the ranking of the retrieved images and avoid false-positives. As this provision is not modelled by AUC-PR and presents an important case study, this metric has been included in VPR-Bench.

Limitations: There may be cases where false-positive rejection is not possible, e.g., geometric-verification may fail in dark, unstructured environments and in extreme conditions (rain, fog, etc), and therefore in such cases it may be relevant to use VPR systems (and metrics like AUC-PR) that are highly precise and where the best matched image should *not* be a false-positive. On the other hand, similar to AUC-PR, RecallRate@N also rewards a VPR system only for retrieving a single correct match per query from the reference database. Neither the metrics penalize nor reward retrieval of more than one correct match per query, which is a particular use-case for the mean-Average-Precision (mAP) metric.

Metric design: The requirement for RecallRate@N is that the correct reference image for a query only needs to be among the Top-N retrieved images. Let the total number of query images with a correct match among the Top-N retrieved images be M_Q, and the total number of query images be N_Q, then the RecallRate@N can be computed as

$$RecallRate@N = \frac{M_Q}{N_Q}. \tag{3.3}$$

Please note that RecallRate@1 is actually equal to the Precision at maximum Recall P_{Rmax}. The ideal value of RecallRate@N is equal to 1. RecallRate@N does not consider false-negatives (incorrectly discarded correct matches) and true-negatives (new places) and is therefore not a replacement for AUC-PR and AUC-ROC, respectively. An ideal RecallRate@N graph should represent a straight line on y-axis=1 (RecallRate=1) for all values of N on the x-axis.

ROC CURVES

Motivation: AUC-PR and RecallRate@N do not consider true-negatives within them. In VPR, true-negatives are those query images for which the ground-truth correct reference match does not exist. These true-negatives can also be thought of as 'new places', i.e., places which haven't been seen before by the vision system. It is important for a VPR system to identify these true-negatives for their usage within a topological SLAM system for an exploration task. Previous metrics like AUC-PR and RecallRate@N are designed for tasks where a map is already available and the primary task of the VPR system is only accurate localisation. AUC-ROC, therefore, complements the analysis provided by AUC-PR and/or RecallRate, but does not replace them.

Limitations: ROC curves are useful for balanced class problems and therefore in datasets where true-negatives and true-positives are not balanced, ROC curves may not present value. ROC curves are also not useful for applications that already have a fixed map of the environment available, because in this case, identification of new places is not a requirement.

Metric design: In order to assess the true-negative classification performance of a VPR system, the well-established Receiver-Operating-Characteristic (ROC) curve is utilized.

Because VPR datasets in general do not contain any true-negatives, they represent an imbalanced class problem, i.e., true-positives and true-negatives classes are not balanced. This is another reason why ROC curves have not been used for VPR evaluation, as the focus has always been on achieving very high-precision, i.e. retrieving as many correct place matches as possible. The author therefore manually adds true-negatives to the Gardens Point dataset for ROC evaluation, where true-negatives are images taken from the Nordland dataset as a case-study. The modified Gardens Point dataset contains the 200 original true-positives and the 200 added true-negatives from the Nordland dataset. The reference database remains the same, while the ground-truth is modified such that for the 200 true-negative query images, it identifies that a correct match does not exist. This modified dataset and associated ground-truth is available separately in the framework to avoid confusion with the original datasets. It is easily possible to extend this analysis to other datasets, and it is supported by the proposed framework.

The definitions of true-positives, false-positives, and false-negatives for ROC curves remain the same as PR curves, with only the extra addition of true-negatives as defined above. An ROC curve is a plot between the true-positive rate (TPR) on the vertical axis and the false-positive rate (FPR) on the horizontal axis. The TPR signifies how many of the total query images that have a correct reference match have been retrieved by a VPR technique. The FPR identifies how many of the total query images that do not have a correct reference match were labeled as false-positives. These metrics are computed as

$$TPR = \frac{True\ Positives}{True\ Positives + False\ Negatives}, \tag{3.4}$$

$$FPR = \frac{False\ Positives}{False\ Positives + True\ Negatives}. \tag{3.5}$$

Similar to PR-curves, the true-positive rate and the false-positive rate are computed for a range of different matching confidence thresholds. Area under this ROC curve (AUC-ROC) is used to model the classification quality of a VPR technique. A perfect AUC-ROC is equal to 1, and an ideal ROC curve is identified by TPR=1 for all values of FPR. An AUC-ROC of 0.5 identifies that a technique has no separation capacity between the true-class (queries with existing matches in the reference database) and the false-class (new places). An AUC-ROC below 0.5 means that a technique is yielding opposite labels for most of the candidates, i.e., true-positives are classified as true-negatives and vice-versa.

IMAGE RETRIEVAL TIME

Motivation: From a computational perspective, the most important factors to consider are the feature encoding time and the descriptor matching time of VPR techniques, which have been usually reported by works from both the VPR communities. These computational metrics only complement the metrics related to place matching precision. In applications where the reference database is significantly large[2], descriptor matching time may be more relevant than feature encoding time and vice-versa.

[2]The quantified meaning of 'large' is usually dependent upon the computational platform, system's implementation, and the ratio of feature encoding time to descriptor matching time.

Limitations: Unlike other precision-related metrics, computational performance is greatly dependent on the underlying platform and can change significantly from one system to another.

Metric design: Feature encoding time and descriptor matching time can be combined together to model the image retrieval time of a given VPR technique. Let the total number of images in the map (reference database) be Z. Let t_e represent the feature encoding time and t_m represents the time required to match feature descriptors of two images. Also, let the retrieval-time of a VPR technique be denoted as t_R, where this t_R represents the time taken (in seconds) by a VPR technique to encode an input query image and match it with the total number of images (Z) in the reference map to output a potential place matching candidate. This t_R is modelled as

$$t_R = t_e + O(Z) \times t_m. \tag{3.6}$$

Here $O(Z)$ represents the complexity of search mechanism for image matching and could be linear, logarithmic, or other, depending upon the employed neighbourhood selection mechanism (e.g., linear search, nearest-neighbour search, approximate nearest neighbour search, etc.). While implementing this framework, the author ensured that t_e and t_m are computed in a fashion where all subsequent dependencies, input/output data transfer, pre-processing, and preparations of a VPR technique are included in these timings for a fair comparison. The descriptor matching time is related to the descriptor size, computational platform, descriptor dimensions, and descriptor data-type, which have all been reported in this chapter for completeness.

In addition to the metrics discussed previously, the feature descriptor size of all VPR techniques is computed and reported to reflect the storage requirements, which are highly relevant for large-scale maps.

TRUE-POSITIVES DISTRIBUTION ANALYSIS

Motivation Some robotics applications may require that a loop-closure candidate (a correct VPR match) must be obtained at least every Y meters over a traversed trajectory. For a robot localisation system (visual-inertial-based, visual-SLAM-based, dead-reckoning-based and similar), a VPR technique that is moderately precise but has a uniform true-positive distribution over the robot's trajectory has more value than a highly-precise technique with a non-uniform distribution. The author has therefore included true-positives distribution over trajectory analysis in the proposed benchmark.

Limitations: This metric is application-specific and does not provide insights for the non-traversal datasets usually employed by the computer vision VPR community.

Metric design: This metric was presented by [141]. They analyse the distribution of loop-closure candidates (true-positives) by creating histograms identifying inter-loop-closure distances, such that the height of the histogram bar specifies the number of loop-closures performed in the dataset with that particular inter-frame distance constraint. The same analysis schema is used in this work.

OTHER VPR METRICS

The metrics discussed previously in this chapter have their specific utilities, and in some cases these metrics complement each other (e.g., AUC-PR and RecallRate@N), and in other cases, provide dedicated value (e.g., AUC-ROC for true-negatives, retrieval time

for computational analysis). Still, even more metrics have been used for VPR, including mAP ([96]), Performance-per-Compute-Unit ([75], [205]), Recall@0.95 Precision ([206], Extended Precision ([142]), F1-score ([140]), error-rate ([207]) and Recall@100% Precision ([76]). To limit the scope of the analysis performed in this chapter, and because there is a high correlation between some of these metrics (e.g., between RecallRate@N, Recall@100% Precision, and Recall@95% Precision), the author has implemented many of these other metrics in the implementation of VPR-Bench, but did not include them in this chapter's experiments.

3.2.5 INVARIANCE QUANTIFICATION SETUP

In this sub-section (and its respective results/analysis in sub-section 3.3.8), the author proposes a thorough sweep over a wide range of quantified viewpoint and illumination variations and studies the effect on VPR techniques.

[193] proposed a well-designed and highly-detailed dataset, namely the Point Features dataset, where a synthetically created scene is captured from 119 different viewpoints, under 19 different illumination conditions. While the original dataset consists of different synthetic scenes, some of which are irrelevant to VPR, the author thus utilises a subset of the dataset that represents scenes of synthetically-created 'Places', and the author thus uses 2 of these scenes/places in this chapter. The author has thus integrated this subset of the Point Features dataset in the proposed framework, and sub-section 3.2.5 is dedicated to explaining the details of this dataset.

An obvious limitation of the Point Features dataset is that it depicts synthetic scenes (toy-houses, toy-cars etc) instead of a real-world scene. This limitation is a challenge to address because in real-world scenes, it is significantly difficult to control the illumination of a scene. However, the author does make an effort in this chapter to present the analysis of viewpoint and illumination variation effects on VPR performance for real-world variation-quantified (semi-quantified) datasets as well. The level of quantification available in these datasets is not as detailed as the Point Features dataset, but they serve to bridge the sim-to-real gap in the proposed evaluation to some degree. Therefore, in this reference, the chapter has used the QUT multi-lane dataset ([194]) for viewpoint variations and the MIT multi-illumination dataset ([195]) for illumination variations. Details of both of these datasets are available in their respective sub-sections below.

Section 3.2.5) is dedicated to present the details of the evaluation mechanism on these 3 datasets. The evaluation mechanism in this chapter (and in the proposed framework) is kept the same for all 3 datasets (Point-features, QUT multi-lane, MIT multi-illumination datasets) to ensure consistency. Please note that throughout this chapter, the term 'same-but-varied place' refers to the images of a place from different viewpoints or under different illumination conditions, while the term 'different place' refers to a place that is geographically not the same as the 'same-but-varied' place. For each of the 3 datasets in this section, there are only 2 actual places in total, i.e., 'the same-but-varied' place and the 'different place'.

POINT FEATURES DATASET

The Point Features dataset can be broadly classified to have 3 variations: 1) Viewpoint, 2) Illumination and 3) Scene. The former two variations are fully used in this chapter, while only two relevant scenes (representing two different places) are utilised from the latter.

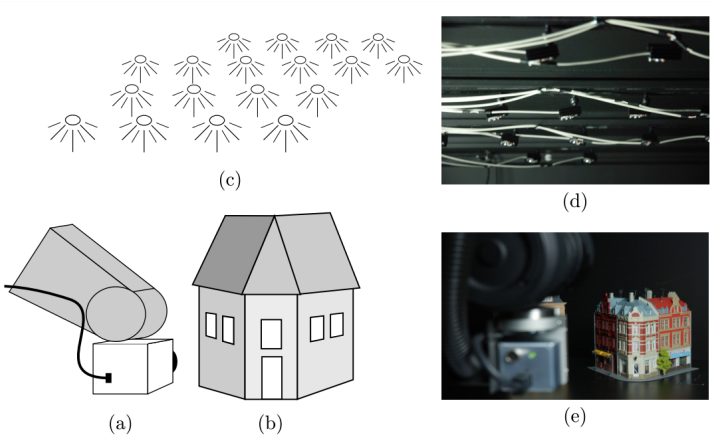


Figure 3.4: The schematic setup of the Point Features dataset has been reproduced here with permission from [193]. The dataset primarily consists of (a) A camera mounted on a robot-arm, (b) Synthetic Scene, (c) LED arrays for illumination, (d) (e) Snapshots of the actual setup.

The authors ([193]) achieve viewpoint-variation by mounting the scene facing camera on a highly-precise robot arm, where this robot arm is configured to move across and in-between 3 different arcs, that amount to a total of 119 different viewpoints, as depicted in Fig. 3.5. Their setup used 19 LEDs that varied from left-to-right and front-to-back to depict a varying directional light source. This directional illumination setup has been reproduced in Fig. 3.6, while the azimuth (ϕ) and elevation angle (θ) of each LED is listed in Table 3.4. Fig. 3.4 shows various components of the dataset, while in Fig. 3.7 the author qualitatively shows all the 19 different illumination cases on one of the scenes.

QUT MULTI-LANE DATASET

The QUT multi-lane dataset is a small-scale dataset depicting a traversal through an outdoor environment ([194]) performed at 5 different laterally-shifted viewpoints under similar illumination and seasonal conditions. This traversal has been performed at a near-constant velocity by a human from an ego-centric viewpoint. The dataset contains 2 types of viewpoint changes: (a) Forward and Backward movement, i.e., Zoom-in and Zoom-out effect similar to the inter-arc viewpoint change of the Point Features dataset, (b) Lateral viewpoint change, which is close to the viewpoint change across the arcs of the Point Features dataset.

The author uses in total 2 different scenes (representing 2 different places) from their traversal, and for each scene uses 15 viewpoints. These 15 viewpoints represent 5 lateral viewpoint changes for 3 consecutive (forward/backward movement) viewpoints of each scene/place. The lateral viewpoint change is almost 1.2 meters, while the forward/backward viewpoint change is around 3.5 meters. Examples of these viewpoint changes have been shown in Fig. 3.8 for both the scenes/places.

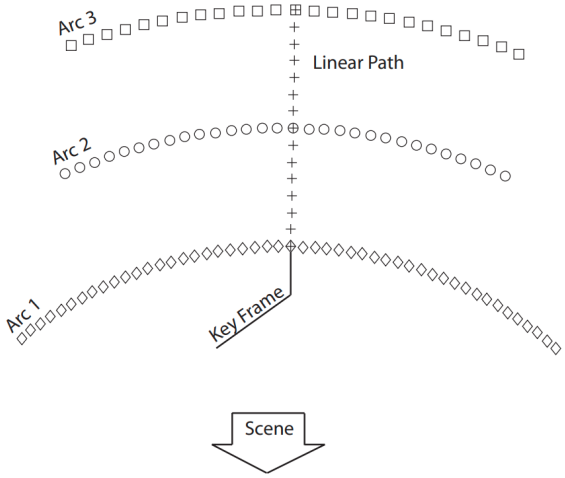


Figure 3.5: The 119 different viewpoints in the Point Features dataset have been reproduced here with permission from [193]. Camera is directed towards the scene from all viewpoints. Arc 1, 2 and 3 span 40, 25 and 20 degrees, respectively, while the radii are 0.5, 0.65 and 0.8 meters.

MIT MULTI-ILLUMINATION DATASET

The MIT multi-illumination dataset was proposed by [195]. This dataset represents a variety of indoor scenes captured under 25 different illumination conditions. Most of the scenes represented in this dataset may not actually be classified as 'Places', however because only two scenes/places are required, the author has manually mined scenes that represent an indoor appearance of a place and are feature-full. [3]

The dataset consists of a total of 1016 interior scenes, each photographed under 25 predetermined lighting directions, sampled over the upper hemisphere relative to the camera. All of these scenes depict common domestic and office environments. The scenes are also populated with various objects, some of which represent shiny surfaces and are therefore interesting for our evaluation. The lighting variations are achieved by directing a concentrated flash beam towards the walls and ceiling of the room, which is similar to the works of [208] and [209]. The bright spot of light that bounces off the wall becomes a virtual light source that is the dominant source of illumination for the scene in front of the camera. The approximate position of the bounce light is controlled by rotating the flash head over a standardized set of directions. The authors of MIT multi-illumination dataset propose that their camera and flash system is more portable than dedicated light sources, which simplifies its deployment 'in the wild'. Because the precise intensity, sharpness, and direction of the illumination resulting from the bounced flash depends on the room geometry and its materials, these lighting conditions have been recorded by inserting a pair of light probes, a reflective chrome sphere,

[3]The author acknowledges that even the multi-illumination dataset may not fully represent a real-world 'landmark' and multiple illumination sources, etc, however, to the best of the author's knowledge, this is the most relevant real-world illumination quantified dataset for the problem at hand. Controlled illumination, especially in outdoor scenes is notoriously difficult as identified by [195].

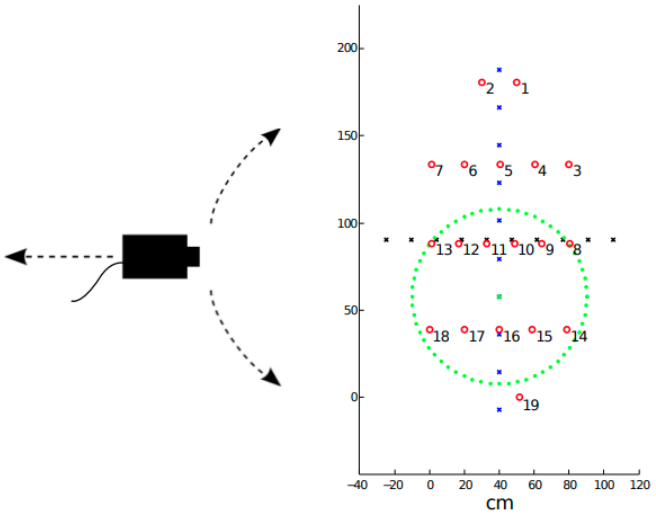


Figure 3.6: The distribution of LEDs across physical space is shown as seen from above. Each red circle represents an LED and only a single LED is illuminated at a point in time, yielding 19 different illumination conditions. In the original work, [193], used artificial linear relighting from left-to-right (blue) and front-to-back (black) based on a Gaussian-weighting, as depicted with the green-circle, but in this chapter only the original 19 single-LED illuminated cases are used. These 19 cases (red-circles) need to be seen in correspondence with Table 3.4.

and a plastic gray sphere, at the bottom edge of every image. For further specification details, I would refer the reader to the original paper of [195] for avoiding textual redundancies. Examples of the 2 different places under the varying illumination conditions have been shown in Fig. 3.9, where Place 1 is chosen due to its closest-possible depiction of an indoor VPR-relevant scene, while Place 2 is chosen due to the shiny objects in that scene. Both the scenes/places are feature-full.

EVALUATION MECHANISM

In order to utilise the densely-sampled viewpoint and illumination conditions in the Point Feature dataset (and the less-detailed QUT multi-lane dataset and the MIT multi-illumination dataset), an evaluation scheme was needed where VPR performance variation could be quantified and analyzed. This quantification is not possible with the traditional place matching evaluation, where there are only two possible outcomes for a given query image, i.e. a correct match or a false match. This is because the mismatch cannot be guaranteed to have resulted from that particular variation and may have resulted from perceptual-aliasing or a smaller map size. Also, even if an image is matched, it is not guaranteed that increasing the map-size (i.e., the no. of reference images) would not affect the outcome, as the greater the number of reference images, the greater the chances of mismatch. However, each VPR technique does yield a confidence-score for the similarity of two images/places. Ideally, if two images represent the same place, then the confidence-score should remain the same, if one of the image of that place is varied with respect to viewpoint or illumination, while keeping the other constant. However, in practical cases, VPR techniques are not

Table 3.4: The azimuth (ϕ) and elevation angles (θ) of each LED are listed here (in degrees) with respect to the physical table surface that acts as the center of coordinate system.

LED Number	θ	ϕ	LED Number	θ	ϕ
1	264	57	11	28	86
2	277	57	12	10	80
3	227	68	13	6	74
4	245	72	14	125	65
5	270	73	15	109	68
6	297	72	16	89	69
7	314	68	17	69	68
8	174	74	18	53	64
9	170	80	19	97	56
10	152	86			

fully-immune to such variations, and a useful analysis would be to see this effect on the confidence-score.

Therefore, my analysis on the 3 datasets in this chapter and the VPR-Bench framework are developed based on the effect of viewpoint- and illumination-variation on the confidence score. This confidence score usually refers to the matching score (L1-matching, L2-matching, cosine-matching etc.) in VPR research, and for two exactly similar images (i.e., two copies of an image), this confidence/matching score is always equal to 1. However, when the image of the same place/scene is varied with respect to viewpoint or illumination, the confidence score decreases. This decrease in matching score by varying images of the same place/scene along the pre-known, numerically-quantified viewpoint- and illumination-levels of the 3 datasets presents analytically and visually the limits of invariance of a VPR technique. However, the trends of these variations in-between different VPR techniques cannot be compared solely based on the decrease of confidence scores, due to different matching methodologies. Therefore, for each VPR technique, the author plots the confidence score variation trend for the same place along with the trend for a different place/scene. The point at which the matching score for the same place (but viewpoint- or illumination-varied) approaches near (or below) the matching score for a different place, identifies the numeric value of viewpoint/illumination change that a VPR technique cannot prospectively handle within the test setup.

Evaluation mechanism Point Features dataset: There are a total of 119 different viewpoint positions and 19 different illumination levels. The illumination case 1 in Fig. 3.6 and the left-most point on Arc 1 of Fig. 3.5 are considered as the keyframe(s) for viewpoint- and illumination-invariance analysis, respectively. The 119 viewpoint positions are numerically labelled in consecutive ascending order from the keyframe (labelled as '1') to the right-most point on Arc 1, followed by the leftmost point on Arc 2 to the rightmost point on Arc 2, which is then followed by the left-most point on Arc 3 and the last (labelled as '119') position is the right-most point on Arc 3. For each analysis and each VPR technique, the key-frame is matched with itself to provide an ideal matching score, i.e. 1. For viewpoint-variation analysis, the illumination type/level is kept constant, then Arc 1 is traversed in a

Figure 3.7: The change in appearance of a scene for 19 different illumination levels is shown here from the Point Features dataset.

clock-wise fashion and the author computes the matching scores between the keyframe and the viewpoint-varied (quantified) images. The same is repeated for Arcs 2 and 3, where the keyframe remains the same i.e., the left-most point on Arc 1. The matching scheme yields a total of 119 different matching scores for each of the 119 different viewpoint positions.

For the illumination invariance analysis, the 19 illumination cases are identified numerically in Table 3.4 and qualitatively in Fig. 3.7. For the illumination-invariance analysis, the viewpoint position is kept constant (i.e., the left-most point on Arc 1) and the illumination levels are varied.

Because the decline in matching score itself does not provide too much insight, the matching scores for the same-but-varied scene in the Point Features dataset are plotted along with the matching scores when the reference scene is a different place (i.e. the query/keypoint frame and reference frame are different places). For computing the matching scores between the keyframe and the different scene/place, all of the 119 viewpoint positions and the 19 illumination levels of the different scene/place are utilized. This gives the corresponding number (119/19 for both variations) of data-points for the confidence scores between keyframe and the different place to be plotted against the data-points for the same-but-varied place. There are further advantages to using all the (119 and 19) viewpoint and illumination cases for the different place, as explained later in sub-section 3.3.8.

Evaluation mechanism QUT Multi-lane dataset: The evaluation mechanism is the same for QUT Multi-lane Dataset as that for the Point Features dataset. In this case, however, there are a total of 15 different viewpoint positions for the same-but-varied place and 15

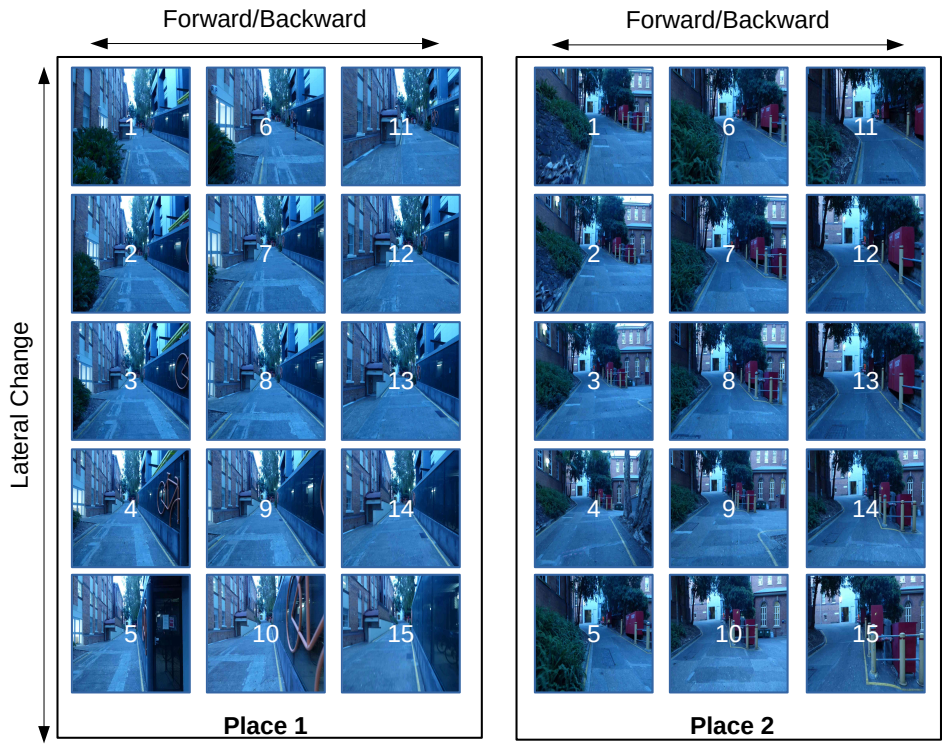


Figure 3.8: The 15 different viewpoint cases in the QUT multi-lane dataset for both the scenes/places have been presented here.

different viewpoint positions for the different place. Unlike the large number of viewpoint variations in the Point Features dataset which were difficult to qualitatively represent, the 15 different viewpoint positions for both the scenes/places for the QUT multi-lane dataset have been shown and labelled in Fig. 3.8. For both the scenes/places, the viewpoint positions 1-5 are left-to-right variations at the beginning of the traversal, 6-10 are left-to-right variations a few meters ahead of 1-5, and 11-15 are left-to-right variations a few meters ahead of 6-10. Image 1 of Place 1 serves as the keyframe. The matching scores between the keyframe and the same-but-varied place, and between the keyframe and the 15 viewpoints of different place (place 2) are computed in the same fashion as that for Point Features dataset.

Evaluation mechanism MIT Multi-illumination dataset: The evaluation mechanism for the MIT multi-illumination dataset is also the same as that of the Point Features dataset. In this case, however, there are a total of 25 different illumination cases. These illumination cases for both the scenes have been identified in Fig. 3.9. Image 1 of Place 1 serves as the keyframe. The matching scores between the keyframe and the same-but-illumination-varied place, and between the keyframe and the 25 different illuminations of different place (place 2) are computed in the same fashion as that for the Point Features dataset.

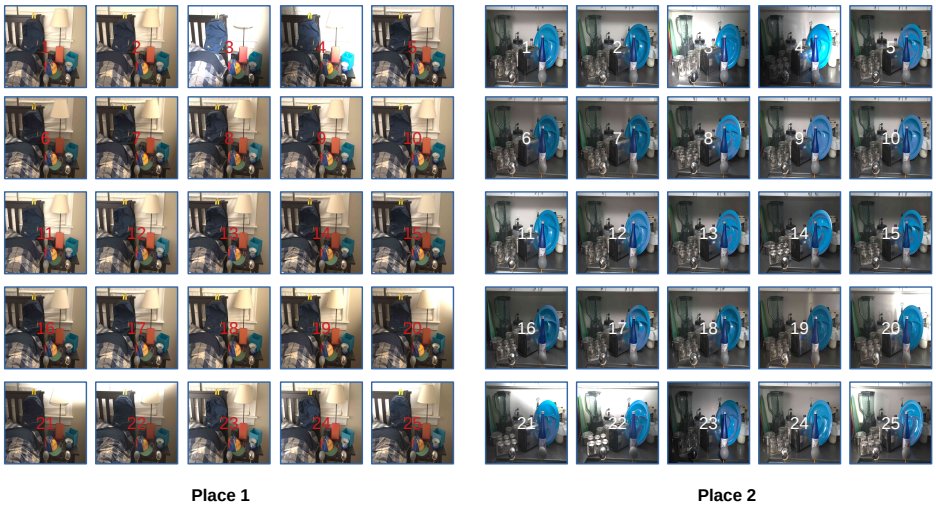


Figure 3.9: The 25 different illumination cases for both the scenes/places from the MIT multi-illumination dataset have been presented here.

3.3 EXPERIMENTS

In this section, the author presents detailed results and analysis for the 10 VPR techniques on the 12 datasets for various evaluation metrics. The variation in performance by varying dataset ground-truths, computational platforms (CPU vs GPU), feature descriptor sizes, and the retrieval timings vs platform speed is discussed. An extensive analysis is provided based on the viewpoint and illumination invariance quantification setup. Finally, the role of viewpoint variance vs invariance and the subjective requirements of these from a VPR system are discussed. The experiments were performed on a Ubuntu 20.04.1 LTS operating system running on an AMD(R) Ryzen(TM) 7-3700U CPU @ 2.30GHz.

Table 3.5: The values of AUC-PR are listed here for all the techniques on the 12 datasets. The bold values in each row represent the state-of-the-art technique for each dataset for the corresponding metric.

Dataset Name	NetVLAD	RegionVLAD	CoHOG	HOG	AlexNet	AMOSNet	HybridNet	CALC	AP-GeM	DenseVLAD
Gardens Point	0.70	0.56	0.42	0.28	0.47	0.57	0.59	0.38	0.67	**0.77**
SPEDTest	0.81	0.61	0.48	0.63	0.63	**0.91**	0.90	0.67	0.71	0.85
Nordland	0.08	0.12	0.02	0.02	0.20	**0.30**	0.17	0.12	0.06	0.13
Living Room	0.94	0.94	0.85	**1.00**	0.95	0.98	0.97	0.70	0.93	0.99
Synthia	0.92	0.60	0.79	**0.99**	0.88	0.89	0.91	0.90	0.97	**0.99**
17Places	0.39	0.38	0.40	0.29	0.39	0.37	0.39	**0.45**	0.36	0.38
Cross-Seasons	**0.99**	0.94	0.72	0.87	**0.99**	0.98	**0.99**	0.71	0.98	**0.99**
Corridor	0.83	0.66	0.69	0.68	0.80	**0.95**	0.93	0.78	0.85	0.89
Tokyo 24/7	0.89	0.42	0.09	0.00	0.06	0.25	0.28	0.01	0.78	**0.95**
ESSEX3IN1	0.71	0.55	0.80	0.09	0.16	0.30	0.32	0.16	0.72	**0.98**
Pittsburgh	0.94	0.73	**0.97**	0.01	0.05	0.08	0.08	0.02	0.86	0.95
INRIA Holidays	0.90	0.94	0.76	0.39	0.79	0.89	0.92	0.77	0.98	**0.99**

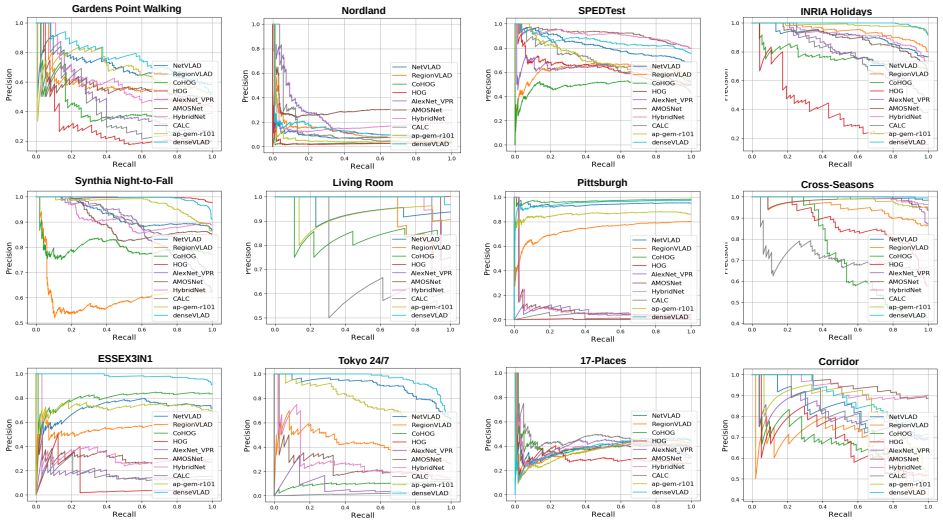


Figure 3.10: The Precision-Recall curves for all 10 VPR techniques generated on the 12 datasets by VPR-Bench framework are presented here.

3.3.1 PLACE MATCHING PERFORMANCE

The results obtained by executing the VPR-Bench framework, given the attributes presented in Section 3.2, are now presented.

PR-curves: Firstly, the precision-recall curves for all 10 VPR techniques on the 12 indoor and outdoor datasets are presented in Fig. 3.10. The values of AUC-PR for all techniques have been listed in Table 3.5. From the perspective of place matching precision, VPR-specific deep-learning techniques generally perform better than non-deep-learning techniques, with the exception of CoHOG and DenseVLAD, which always performs better than AlexNet and CALC. While CoHOG can handle lateral viewpoint-variation, it cannot handle 3D viewpoint-variation as present in the Tokyo 24/7 dataset. NetVLAD and DenseVLAD can handle 3D viewpoint-variation better than any other technique, because the training dataset for these contained 3D viewpoint-variations. HybridNet and AMOSNet can handle only moderate viewpoint-variations, but perform well under conditional variations due to training on highly conditionally-variant SPED dataset. Please note that the SPED dataset and SPEDTest dataset do not contain the same images, therefore the state-of-the-art performance of HybridNet and AMOSNet on SPEDTest dataset advocates for the utility of deep-learning techniques in environments similar to training environments (which in this case is the world from a CCTV's point-of-view).

All techniques suffer on the Nordland dataset, which contains significant perceptual aliasing and a large reference database. HOG and AlexNet usually lie on the lower-end of matching capabilities for all viewpoint-variant datasets, but perform acceptably on moderately condition-variant datasets that have *no* viewpoint variation. A notable exception here is the state-of-the-art performance of HOG compared to all other techniques on the Living

Table 3.6: The values of feature encoding time t_e (sec), descriptor matching time t_m (msec) are listed here for 8 VPR techniques. Encoding time is dependent upon the image resolution, however, in this chapter, I have used the recommended image resolutions by the authors of the respective VPR techniques, and therefore t_e is independent of the underlying dataset. The second row reports t_m for the techniques' default data-types as given in the 6th row, while the values of t_m in the third row are for the fixed float-64 data-type of descriptors for all techniques. Please see the accompanying text regarding trends in the descriptor matching time. The 4th row shows feature descriptor sizes of all 8 VPR techniques in Kilo-Bytes (KBs) for a single image, along with the descriptor dimensions and default data-types in the following rows. The bold values in each row represent the state-of-the-art technique for the corresponding metric. Because DenseVLAD and GeM results have been computed using a different computational platform, the values for these techniques have not been included here to keep the comparison fair.

Metric	NetVLAD	RegionVLAD	CoHOG	HOG	AlexNet	AMOSNet	HybridNet	CALC
t_e	3.71	1.29	0.06	**0.007**	1.14	0.80	0.81	0.04
t_m (default)	0.06	0.17	2.64	0.07	0.03	0.13	0.13	**0.02**
t_m (float-64)	0.08	0.17	6.91	0.49	**0.04**	0.13	0.13	**0.04**
Desc. Size (KBs)	16.38	786	123	138.38	8.51	61.4	61.4	**4.25**
Desc. Dimensions	1×4096	256×384	32×961	1×34596	1×1064	256×30	256×30	1×1064
Data Type	float-32	float-64	float-32	float-32	float-64	float-64	float-64	float-32

Room dataset, which consists of high-quality images of places under indoor illumination variations. This suggests that on very small-scale datasets (and therefore for such small-scale indoor robotics applications), simple handcrafted techniques can yield good matching performance even under moderate variations in viewpoint and illumination. CALC cannot handle conditional variations to the same level as other deep-learning-based techniques, as the auto-encoder in CALC is only trained to handle moderate and uniform illumination changes. Region-VLAD also performs in the same spectrum as NetVLAD, but cannot surpass it on most datasets. All techniques perform poorly on the 17 Places dataset that represents a challenging indoor environment with strict viewpoint variance, suggesting that the outdoor performance success of techniques cannot be extended to an indoor environment. The perceptual-aliasing of datasets like Cross-Seasons and Synthia also presents significant challenges to VPR techniques. The AUC-PR of HOG comes out as 1 for the Living Room dataset, because a threshold exists above which all images are correct matches (17 out of 32) and below which (15 out of 32) all images are incorrect matches. The results on Pittsburgh dataset and Tokyo 24/7 dataset identify two very separable clusters of VPR techniques: those (e.g. AMOSNet, HybridNet, CALC) that cannot handle large reference databases, which essentially have many distractors, and those (e.g. NetVLAD, DenseVLAD, CoHOG) which can handle such large reference databases.

RecallRate@N: While for AUC-PR, the results have been listed in Table 3.5, RecallRate@N is usually represented as a trend and not as a single value. Therefore, for RecallRate@N, the variations in RecallRate for values of N in the range of 1 to 20 are plotted. These plots have been created for all the 10 VPR techniques on the 12 datasets and are shown in Fig. 3.11. Clearly, increasing/relaxing the value of N leads to an increase in RecallRate for all 10 techniques and thus systems/applications that have a subsequent verification stage to re-rank the output of a VPR system would benefit from the trends presented in Fig. 3.11. An interesting insight is depicted by the values of N on which the ordering of techniques changes, which re-affirms the utility of this metric, for example, see results on Gardens Point, ESSEX3IN1, Cross-Seasons, and Corridor datasets. CALC starts from the bottom for RecallRate@1 on the Living Room dataset and sharply rises

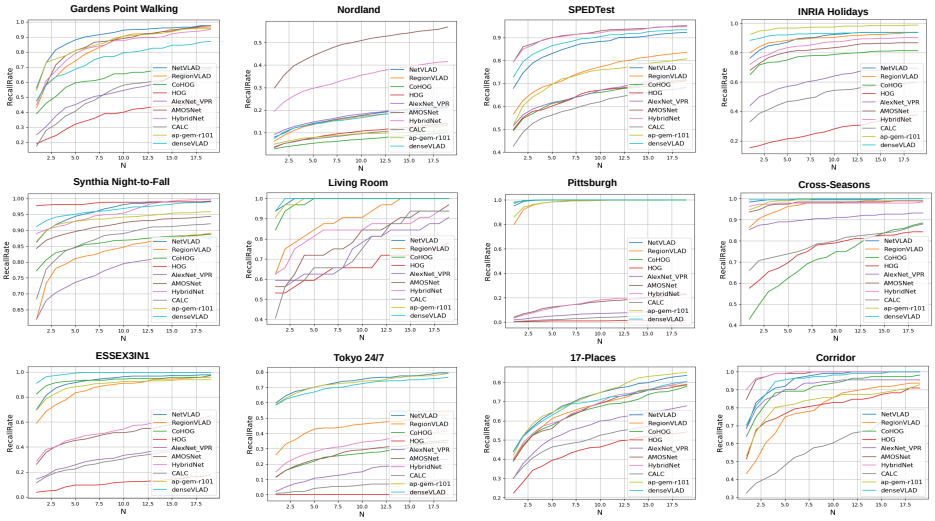


Figure 3.11: The RecallRate@N curves for all 10 VPR techniques generated on the 12 datasets by VPR-Bench framework are presented here. The range of N used here is 1 to 20 with a step-size of 1. The values of RecallRate@1 represent the Precision@100% Recall of a VPR technique.

for later values of N. It is important to note the changing state-of-the-art for RecallRate in comparison to AUC-PR, for example, DenseVLAD is the state-of-the-art on Tokyo 24/7 dataset for AUC-PR, but for most values of RecallRate, NetVLAD and AP-GeM outperform DenseVLAD. Examples of images matched/mismatched by all VPR techniques on the 12 datasets are shown in Fig. 3.12 for a qualitative insight.

Computational performance: The values of feature encoding time, descriptor matching time and descriptor size have been listed in Table 3.6 for the author's fixed platform. For all experiments in this work, the default data-types of descriptors are used, as specified in Table 3.6 last row, however, for the sake of complete comparison of matching time t_m, the data-types of all techniques is affixed to float-64 for the values of t_m in Table 3.6 third row. The encoding time is usually higher for deep-learning-based techniques, while the matching time is generally higher for larger feature descriptors. Evidently, there are four factors affecting descriptor matching time: distance/similarity function, number of descriptor dimensions, length of each dimension, and the descriptor data-type. For the reported 64-bit platform, cosine-distance as a similarity function and float-32 data-type, the change of size of a descriptor dimension (e.g., NetVLAD vs HOG in Table 3.6 second row) has less effect on the matching time than a change in the total number of dimensions of a descriptor (e.g., NetVLAD vs CoHOG in Table 3.6 second row). On the other hand, for float-64 data-type and fixed similarity function, the increase in matching time is almost linear with increasing size of a descriptor dimension (e.g., NetVLAD vs HOG in Table 3.6 third row). AMOSNet has half the descriptor size of CoHOG, both descriptors are 2-dimensional, but the matching time for CoHOG is significantly higher than AMOSNet due to different distance functions, i.e., direct L1-distance for AMOSNet and convolutionally applied cosine-distance for CoHOG.

Some of the key findings from the analysis in this sub-section can be summarised as follows:

1. Unlike previous evaluations ([210], [211]), where state-of-the-art AUC-PR performance was almost always achieved by NetVLAD, this chapter shows that state-of-the-art AUC-PR performance is widely distributed among all the techniques across the 12 datasets.

2. The state-of-the-art technique for a particular dataset is metric-dependent and therefore, application-specific. A computationally-restricted application may find metrics like descriptor-size or retrieval-time important, while computationally-powerful platforms may only utilise AUC-PR and RecallRate.

3. Interestingly, hand-crafted and non-deep-learning place recognition techniques can also achieve state-of-the-art performance. For DenseVLAD, this had been previously reported by [73] and [74], and I re-affirm their findings here. In this chapter, the author also shows how HOG and CoHOG have achieved state-of-the-art performance for all metrics on at least one dataset (see results on the Synthia Night-to-Fall dataset and the Pittsburgh dataset in Table 3.5).

4. Applications where the explored environment is small (e.g, a home service robot as in the Living Room dataset) and the variations are moderate, it is better to use a handcrafted computationally-efficient technique, as suggested by results in Table 3.5 for the Living Room dataset.

5. Learning-based techniques that are trained on feature-full datasets do not extend well to non-salient, perceptually-aliased, and feature-less environments. See for example, the matching results on the Nordland dataset and Corridor dataset in Fig. 3.11 and Table 3.5.

6. Because state-of-the-art performance is distributed across the entire set of VPR techniques, an ensemble-based approach presents more value to VPR than a single-technique-based VPR, provided that the high computational and storage requirements of an ensemble can be afforded.

7. A perfect AUC-PR score (i.e., equal to one) may be misinterpreted as a technique that retrieves correct matches for all the query images in the dataset. However, a perfect AUC-PR in fact only means that when the query images and their retrieved matches are collectively arranged in a descending order based on confidence scores, all the true-positives lie above all the false-positives. Thus, it is important that the RecallRate@N (for some value of N) of VPR techniques is also reported in addition to AUC-PR. See, for example, the AUC-PR and RecallRate@1 of HOG on the Living Room dataset, where the former proposes perfect VPR performance while the latter shows a significant room for improvement.

8. The descriptor size of techniques is also a key evaluation metric to be considered. A large descriptor size not only translates into excessive storage needs for the respective reference maps, but also affects the descriptor matching time and leads to higher

Table 3.7: The values of AUC-ROC achieved by 10 VPR techniques on the modified (true-negative added) version of the Gardens Point dataset have been reported here.

NetVLAD	RegionVLAD	CoHOG	HOG	AlexNet	AMOSNet	HybridNet	CALC	AP-GeM	DenseVLAD
0.77	0.64	0.60	0.31	0.70	0.74	0.74	0.82	0.87	0.82

run-time memory (RAM) consumption/needs. The analysis on this is further presented in sub-section 3.3.4.

3.3.2 ROC CURVES: FINDING NEW PLACES

Next, the ROC curves for all techniques on a modified version of the Gardens Point dataset are shown. The Gardens Point dataset has been modified to contain 200 queries as true-negatives in addition to its existing 200 true-positives. The number of true-positives and true-negatives is kept equal, because ROC curves work well for balanced classification problems. These curves have been shown in Fig. 3.13. The author notes that unlike the PR-curves for the techniques on Gardens Point dataset, where most techniques perform very well, the class separation capacity (ROC performance) of these techniques is not as good. However, among the techniques, learning-based techniques clearly outperform handcrafted VPR techniques. Although CALC cannot perform well among learning-based techniques for PR curves, the ROC curves show that it has a better class separation capacity than most of the other learning-based techniques. The AUC-ROC for all the techniques has also been listed in Table 3.7, and all techniques generally achieve a lower AUC-ROC than ideal. The AUC-ROC of HOG is less than 0.5, because it yields opposite labels for true-positives and true-negatives (i.e., existing places are classified as new places and vice versa).

3.3.3 COMPUTATIONAL PERFORMANCE: CPU VS GPU

While the previous sub-sections have shown the performance of 10 VPR techniques on the fronts of place matching precision and computational requirements, the underlying hardware has been a CPU-only platform. Generally, CPU represents the common computational hardware for resource-constrained platforms, but learning-based techniques are favored well by GPU-based platforms. Thus, depending on the underlying platform characteristics (CPU vs GPU), it may or may not be fair to compare handcrafted VPR techniques with deep-learning-based VPR techniques on the computational front.

The author here reports the feature encoding time t_e and the descriptor matching time t_m of the 7 deep-learning-based techniques in VPR-Bench when implemented on a GPU-based platform. The GPU-based evaluation was performed using an Nvidia GeForce GTX 1080 Ti with 12GB memory, using a batch size of 1. The mechanism for computation of the timings is the same as that for CPU (i.e. averaged over the entire dataset), and the same codes/parameters were used as those for CPU. These timings are reported in Table 3.8 for the Gardens Point dataset.

It can be observed that the GPU-based ordering of methods is mostly similar as their CPU-based ordering (see Table 3.6), with notable exception of RegionVLAD vs NetVLAD for t_e, because of the former's compute-intensive CPU-based region-extraction and VLAD description. In general, the computation times between CPU and GPU vary noticeably for

Table 3.8: The values of encoding times and matching times for 7 VPR techniques on the Gardens Point dataset for a GPU-based platform have been reported here.

VPR Technique	t_e (seconds)	t_m (milliseconds)
NetVLAD	0.075	0.002
RegionVLAD	0.451	0.061
AMOSNet	0.032	0.038
HybridNet	0.032	0.035
CALC	**0.001**	**0.001**
AP-GeM	0.027	0.045
AlexNet	0.203	**0.001**

all the methods. This cross-analysis highlights the varying utility of VPR techniques across different platforms.

3.3.4 DESCRIPTOR SIZE ANALYSIS

In this sub-section, the author further extends upon the descriptor size analysis and shows that changing the descriptor size affects various performance-related aspects of a VPR technique, in particular memory footprint, place matching precision, and descriptor matching time. To perform this analysis, the Gardens Point dataset is used, and the various descriptor-related parameters of the 5 VPR techniques are changed, namely CoHOG, HOG, NetVLAD, DenseVLAD, and AP-GeM, which directly affect the descriptor size.

For HOG and CoHOG, the cell-size of the HOG-computation scheme is changed, where the block-size remained twice of the cell-size and all the other parameters like image-size and bin-size were kept constant. For NetVLAD, DenseVLAD and AP-GeM, the PCA output dimensions are changed while all other parameters were kept constant. The effect of these descriptor size changes on the memory footprint (descriptor size), AUC-PR, and descriptor matching time is reported in Table 3.9. The absolute and relative variation of these different performance indicators by changing descriptor size is dependent upon the underlying matching scheme and descriptor dimensions, and this variation is therefore not constant between the different VPR techniques. However, there is a general trend where increasing the descriptor dimension leads to increased descriptor matching time and memory footprint, while AUC-PR also varies for VPR techniques.

The descriptor matching time usually decreases by varying parameters that lead to a decrease in descriptor size. The change in AUC-PR by varying descriptor dimensions is subject to the intrinsics of the individual VPR techniques and the role of their corresponding parameters. For deep-learning-based techniques followed by PCA (see NetVLAD and AP-GeM in Table 3.9)), a decrease in descriptor size may or may not lead to a decrease of AUC-PR, because a decreased descriptor size can lead to either the decrease of confusing/non-salient features (e.g., those coming from vegetation, dynamic objects, etc) or distinguishable/salient features and/or a combination of both. The AUC-PR variation for NetVLAD and AP-GeM generally follows a descending trend with decreasing PCA dimensions, but does remain constant for some immediate steps/levels of PCA. The learning-based DenseVLAD (albeit not deep-learning-based) suffers significantly from the decreased descriptor size. For CoHOG, the AUC-PR variation is similar to the original findings in Zaffar et al. [75], where increasing cell-size leads to reduced viewpoint invariance and lesser AUC-PR. For HOG, the

Table 3.9: The values of AUC-PR, descriptor size (Kilo-Bytes), and matching time (msec) are reported on the Gardens Point dataset by varying descriptor size-related parameters (cell-size and PCA-dimensions) of VPR techniques. Please note that the computations for AP-GeM and DenseVLAD were done on a platform different from that of NetVLAD, HOG, and CoHOG. The maximum PCA dimensions given the AP-GeM default design are 2048.

CoHOG				HOG				NetVLAD				DenseVLAD				AP-GeM			
Cell-Size	AUC	KBs	t_m	Cell-Size	AUC	KBs	t_m	PCA	AUC	KBs	t_m	PCA	AUC	KBs	t_m	PCA	AUC	KBs	t_m
8x8	0.47	508	47.0	8x8	0.19	571	0.14	4096	0.69	16.30	0.06	4096	0.77	16.30	0.06	4096	-	-	
16X16	0.42	123	2.64	16X16	0.29	138	0.07	2048	0.69	8.19	0.06	2048	0.69	8.19	0.06	2048	0.67	8.19	0.06
32X32	0.36	28.8	0.18	32X32	0.29	32.4	0.06	1024	0.59	4.09	0.05	1024	0.64	4.09	0.05	1024	0.65	4.09	0.05
64X64	0.30	6.27	0.06	64X64	0.35	7.05	0.05	512	0.59	2.04	0.05	512	0.58	2.04	0.05	512	0.67	2.04	0.05
128X128	0.19	1.15	0.05	128X128	0.33	1.29	0.04	256	0.52	1.02	0.04	256	0.52	1.02	0.04	256	0.64	1.02	0.04
256X256	0.12	0.128	0.03	256X256	0.16	0.14	0.02	128	0.52	0.51	0.02	128	0.33	0.51	0.02	128	0.62	0.51	0.02

increased cell-size (which reduces descriptor size) actually leads to an increase of AUC-PR due to the optimal settings for the traditional fully global HOG-descriptor scheme. The AUC-PR of HOG is highest for cell-size of 64×64 but decreases when the cell-size in either increased or decreased from this optimal setting. Please note that this optimal setting of the cell-size may differ for different datasets, depending on the amount and nature of viewpoint and conditional variations in the dataset.

3.3.5 TRUE-POSITIVES TRAJECTORY DISTRIBUTION

In addition to the image retrieval timings, it is important to look at the distribution of true-positives (loop-closures) within a dataset sequence. Therefore, as explained in sub-section 3.2.4, the author reports in Fig. 3.14 the distribution of true-positives for 6 trajectory-based datasets. The distribution here refers to the number of true-positives (Y-axis) for a given distance (X-axis) between two correctly retrieved frames. For all the datasets, an inter-frame distance of 1 meter is assumed, i.e., true-positives that are assumed to be 5 meters apart represent two correctly-matched query frames that are 5 frames apart. This assumption is required because the exact knowledge of inter-frame physical distance is unavailable for all the datasets, and because the X-axis can be easily scaled up to represent a different inter-frame distance.

Ideally, all techniques should have a single peak value equal to the total number of query images at the vertical axis in Fig. 3.14. For most techniques on all the datasets, the loop-closures are distributed evenly i.e. curves in Fig. 3.14 peak at small values of X-axis. There is a ripple effect that starts from Y-axis and dies towards larger values of inter-frame distance. This ripple effect is more distributed for Gardens Point and Corridor datasets than the other datasets. Thus, for applications such as SLAM where VPR is used in addition to a visual-localisation system, techniques can mostly achieve periodic loop-closure and correct error-drifts. However, these ripples can be catastrophic for VPR-based topological/primary localisation systems ([46]) which rely solely on location estimated through VPR. This analysis is not provided for non-trajectory-type datasets (SPEDTest, INRIA Holidays, etc), because the inter-frame distance is not a valid assumption for these cases.

3.3.6 ACCEPTABLE GROUND-TRUTH MANIPULATION

An important finding from the analysis performed for sub-section 3.3.1 was that the matching performance also varies depending on the ground-truth information in a VPR dataset. It is possible that the ground-truth is slightly modified such that the new ground-truth is usually

acceptable to the reviewing audience, but it also leads to a change of state-of-the-art technique on a particular dataset. For example, the matching performance varies if the query and reference databases are inter-changed (i.e., query folder becomes the new reference folder and reference folder becomes the new query folder), especially for conditionally-variant datasets. This is shown in Fig. 3.15 for the Nordland and Gardens Point dataset. Here, a small section of the Nordland traversal (as used in [82], [210]) is used, containing 1622 query and 1622 reference images, such that the effects of ground-truth manipulation are more prominent, since all the techniques have very low precision on the full traversal. Interestingly, this analysis reveals that for all the VPR techniques, the rise/decline in performance is not necessarily the same in magnitude and direction. Changing ground-truth in this manner is based on the constraint that reference matches for queries are available from a particular conditional appearance (weather, seasons, time, etc) and that this condition is different from that of query images. This is normally the case for most of the robotics-focused VPR datasets and for applications like teach-and-repeat. This analysis assumes the non-existence of the same appearance conditions of a place in query and reference images.

Moreover, in most of the traversal-based VPR datasets, there is always some level of overlap in visual content between consecutive frames. Thus, techniques which are viewpoint-invariant may get benefits if the ground-truth identifies such frames as correct matches. On the other hand, if the ground-truth only considers frame-to-frame matches (i.e., one query frame has only one correct matching reference frame), such viewpoint-invariant techniques may not get the same matching performance (in the form of AUC-PR, RecallRate@N, EP, etc), because their viewpoint invariance will actually lead to false positives. Examples of these consecutive frames with visual overlap are shown in Fig. 3.16. This effect of changing the ground-truth range on the AUC-PR of various VPR techniques for the Gardens Point dataset and Nordland dataset is shown in Fig. 3.17. One could argue that a correct ground-truth must regard such viewpoint-variant images of the same place as true positives, however, a contrary argument exists for applications that utilise VPR as the primary (only) module for localisation, as discussed further in subsection 3.3.9. This sub-section demonstrates that different state-of-the-arts (i.e., top performing techniques) can be created on the same dataset by manipulating the ground-truth information accordingly.

3.3.7 RETRIEVAL TIME VS PLATFORM SPEED

One of the questions that the author wanted to address through this chapter is, 'What is a good image-retrieval time?'. This is important because most VPR research papers that claim real-time performance consider anything between 5-25 frames-per-second (FPS) as real-time. However, there are two important caveats to such performance. Firstly, the retrieval performance for a VPR application depends on the size of the map. It is therefore important that the size of the map is addressed either by presenting the limits for the map-size or by proposing methodologies to affix the map-size. Secondly, the retrieval performance is directly related to the platform speed. A real-time VPR application may require that a place-match (localisation) is achieved every few meters as a robot traverses through an environment. In such a case, the utility of a technique will depend upon the speed of the robot, as the faster the robot moves, the lower the retrieval time that is acceptable. This is modelled as follows.

Let us assume that a particular application requires K frames-per-meter (where K could

be fractional) and that the robot platform moves with a velocity V. Also, let the size of the map (number of reference images) be Z. Then, the required FPS retrieval performance given the values of K and V is denoted as FPS_{req} and computed as

$$FPS_{req} = K \times V. \tag{3.7}$$

The retrieval performance of a VPR technique will depend on the number of reference images and can be denoted as FPS_{VPR}. This FPS_{VPR} has been modelled previously in equation 3.6, such that $FPS_{VPR} = 1/t_R$. Therefore, to understand the limits of real-time performance of a VPR technique given the application requirements (V, K and Z), the retrieval performance of all techniques is plotted along the platform speed for different values of Z in Fig. 3.18, assuming $K = 0.5$ frames-per-meter. The curves for FPS_{VPR} are straight-lines for constant values of Z, and the range of horizontal-axis (Speed V) for which FPS_{VPR} is less than or equal to FPS_{req} represents the range of platform speed (for that map-size) that a technique can handle. The VPR-Bench framework enables the creation of these curves conveniently and, therefore, presents value to address the subjective real-time nature of a technique's retrieval time for VPR.

3.3.8 INVARIANCE ANALYSIS

One of the key aspects of the VPR-Bench framework, as explained in Section 3.2, is the quantification of viewpoint- and illumination-invariance of a VPR technique. In sub-section 3.3.1, the traditional VPR analysis schema was utilised, where datasets are usually classified based on the qualitative severity of a particular variation. However, in this section, the Point Features dataset presented in sub-section 3.2.5 is utilized, given the quantitative information presented in Fig. 3.5, Fig. 3.6, and Table 3.4.

The change in matching score along these arcs is shown in Fig. 3.19 for all the techniques. There is a clear decline in matching scores as the viewpoint is varied both along the arcs and in-between the arcs. A key insight is that moving along the arcs has more effect (negative) on the matching score than jumping between the arcs (i.e., moving towards or away from the scene). From a computer vision perspective, this means that a change in the scale of the world (zooming-in, zooming-out) has lesser effect on matching scores than the change in 3D-content of the scene.

Ideally, the matching scores for the same scene/place should be equal to 1 for the range of variation a technique can handle, and the matching score for a different scene/place should be 0. However, in practice, all techniques give lower than 1 matching scores when two images of a scene have a particular variation in-between them, while giving higher than 0 scores to places that are different. The point at which the matching score for the same-but-varied place is equal to or lower than 'any' of the matching scores for different place, represents the absolute limits for that VPR technique. Please note that the two curves (same-but-varied place and different place) should not be compared point-to-point, but instead point-to-curve, because the matching score for the same-but-varied place should not be less than 'any' of the matching scores for different place. Thus, while it may appear that the two curves for NetVLAD do not intersect under any viewpoint positions, the matching score for the same-but-varied place for positions $110 - 119$ is almost equal to the matching score for a different place at position 0, which will lead to false positives. A conclusive remark from this

viewpoint-variation analysis is that none of the 8 VPR techniques in this chapter is immune to all levels of viewpoint-variation.

Another benefit of having the matching scores curves for different places, in contrast with the same-but-varied place, is that it allows us to compute the Area-between-the-Curves (ABC) for each of the techniques. These values of ABC have been reported for all the techniques. A higher value of ABC represents that a technique can distinguish well between the same-but-varied place and a different place. The ideal value of ABC is equal to the number of variations (x-axis), as the matching score should remain 1 along the entire x-axis in an ideal scenario. Please note that the ABC does not reflect the absolute matching performance of a VPR technique, and should not be compared with AUC-PR/EP/AUC-ROC, because the analysis is only based on two places/scenes.

The analysis of viewpoint-invariance from the synthetic Point Features dataset is extended to the real-world QUT Multi-lane dataset. The analysis scheme is the same for both datasets, and the obtained curves are shown in Fig. 3.20. The curves on the QUT Multi-lane dataset reaffirm the findings from the Point Features dataset, and the trends on both datasets are similar. More importantly, lateral viewpoint changes have been shown to have a greater effect on the place matching confidence score than the forward/backward movement. The scale/level of this (for viewpoint variations on both the Point Features dataset and the QUT Multi-lane dataset) is, however, dependent upon the scene depth and the exact physical movement for lateral and forward/backward changes. Generally, for higher scene-depth, forward/backward movement leads to a lesser change in visual content than lateral variations and therefore has a lesser effect. Very large forward/backward movement (definition of 'very large' is dependent upon the scene depth) may lead to a greater reduction in confidence score than a small change in lateral viewpoint.

A similar analysis is performed for the 19 different matching scores given the 19 quantified illumination variations, as shown in Fig. 3.19. While the 119 different viewpoint positions represented in Fig. 3.5 are intuitive for analysis, the nature and level of illumination change in Table: 3.4 is not obvious. These 19 different cases are presented qualitatively in Fig. 3.7, so that the illumination-variance curves in Fig. 3.19 can be better understood. It can be seen that uniform or close to uniform changes do not have much effect on the matching score. However, directional illumination changes that lead to the partitioning of a scene between highly-illuminated and low-illuminated portions have the most dramatic effect. An interesting insight is that some basic handcrafted VPR techniques (HOG-based) are able to distinguish between the same-but-illumination-varied places and different places, under all 19 scenarios (i.e., no point on the same-but-varied place curve is lower than any point on the different place curve), while contemporary deep-learning-based techniques struggle with such illumination-variation.

The author has extended this illumination-invariance analysis from the Point Features dataset to the MIT Multi-illumination dataset, and the curves on the Multi-illumination dataset are presented in Fig. 3.21. There is a very sharp drop in place matching confidence for illumination cases 3 and 4 for all the VPR techniques, which reaffirms our finding on the Point Features dataset regarding the significantly large effect of directional illumination change (see Fig. 3.9) on the place matching performance. The effect of illumination change on a handcrafted technique, such as HOG, is lower than that on a learning-based technique like CALC on the MIT Multi-illumination dataset, similar to prior observations on the Point

Features dataset; however, this does not generalize to other learning-based techniques. The reported performance declines by varying illumination cases can be potentially combined with illumination-source prediction works ([212], [213]) to predict when a VPR technique might fail and how different VPR techniques could complement each other in these scenarios.

3.3.9 VARIANCE VS INVARIANCE

A generic perception among the VPR research community, as evident from the recent trend in developing highly viewpoint-invariant VPR techniques, is that the more viewpoint-invariant a technique is, the more utility it has to offer. Through this sub-section, the author takes the opportunity to address that this may not always be the case. In fact, viewpoint variance may actually be required in some applications, instead of viewpoint invariance. A key example here is the applications where VPR techniques act as the primary localization module and where there is no image-to-image, epipolar-geometry-based motion estimation (location refinement) module. For example, [214] extends the concept of VPR for precise localization in mining environments. Similar extensions of VPR as the only module for precise localization are possible in several applications, where an accurate geo-tagged image database of the environment exists, e.g., in factory/plant environments or outdoor applications that can afford to create an *a priori* accurate appearance-based metric/topometric map of the environment. For such applications, VPR techniques are required to have viewpoint-variance, so that even if the two images of the same place are viewpoint-varied, the VPR technique can distinguish between them to perform metrically-precise localization. If a viewpoint-invariant technique is utilized in this scenario, the inherent viewpoint-invariance will lead to discrepancies in localization estimates and eventually cause a system failure.

Thus, a key area to investigate within VPR research should be controlled viewpoint-variance. In sub-section 3.3.8, the author presented a methodology to estimate the viewpoint-invariance of a technique, however, there is no control parameter for any technique that could govern and tune its invariance to viewpoint changes. The author believes that this is an exciting research challenge and should be a topic for VPR research in the upcoming years. Nevertheless, the proposal is that both viewpoint-variance and invariance are desirable properties, depending upon the underlying application, and should be regarded/investigated accordingly.

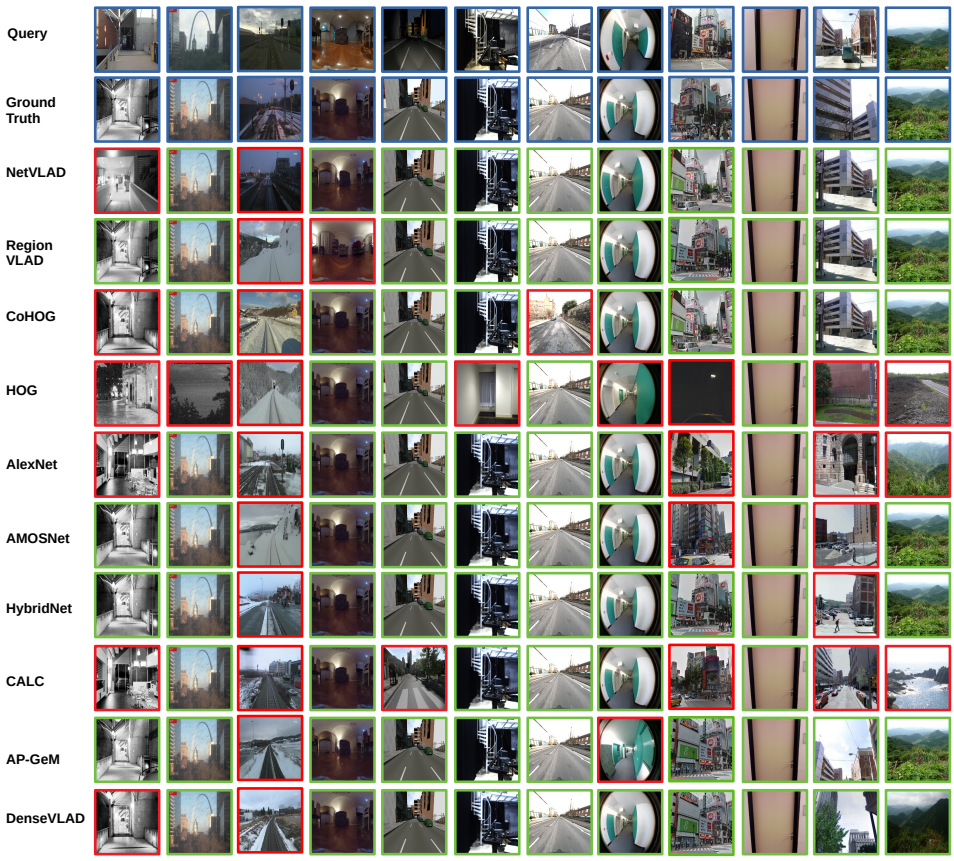


Figure 3.12: Exemplar images matched/mismatched by VPR techniques are shown here for a qualitative insight. Red bounded images are incorrect matches (false positives) and green-bounded images are correct matches (true positives). An image is taken from each of the 12 datasets, where the order of datasets from left to right follows the same sequence as top to bottom in Table 3.5 first column. An important insight here is that some images are matched by all of the techniques, irrespective of the technique's complexities and abilities. This figure also suggests that because almost all of the images are matched by at least 1 technique, an ensemble-based approach can significantly improve the matching performance of a VPR-system.

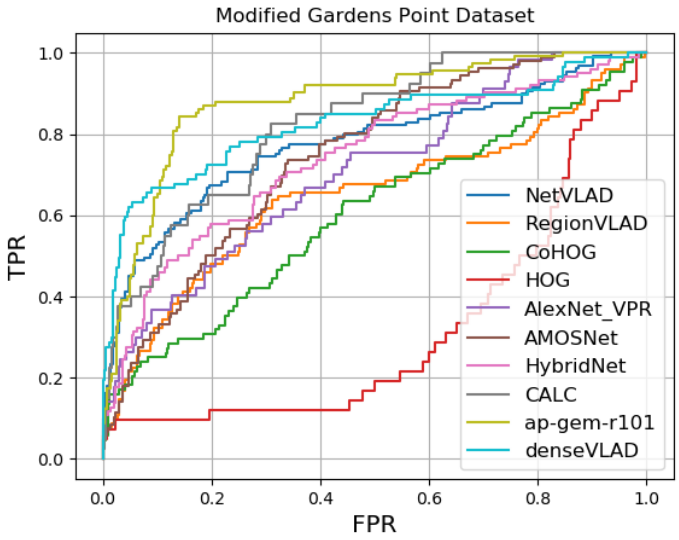


Figure 3.13: The ROC performance of 10 VPR techniques is shown here on a modified (true-negative added) version of Gardens Point dataset that contains 200 true-negatives and 200 true-positives.

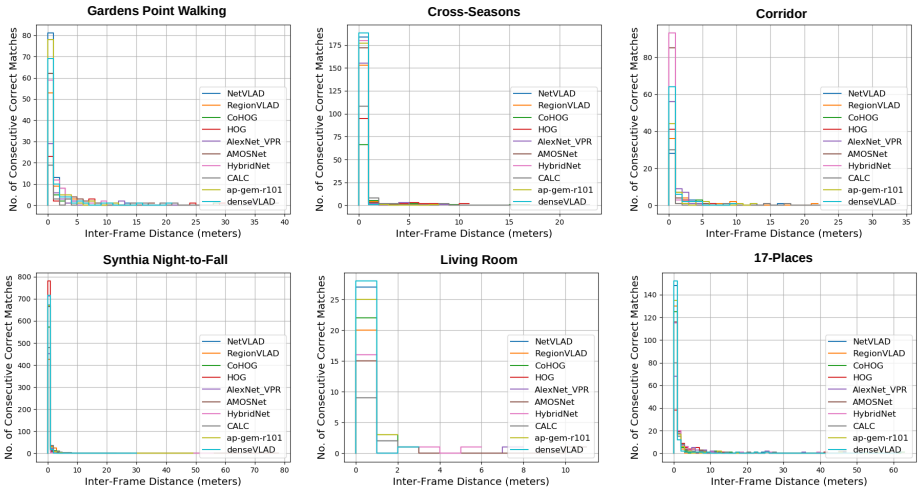


Figure 3.14: The distribution of true-positives over the trajectory of a dataset is shown here. The horizontal axis represents the distance between two consecutive true-positives in a sequence, and the vertical axis shows the number of true-positives that satisfy this distance constraint.

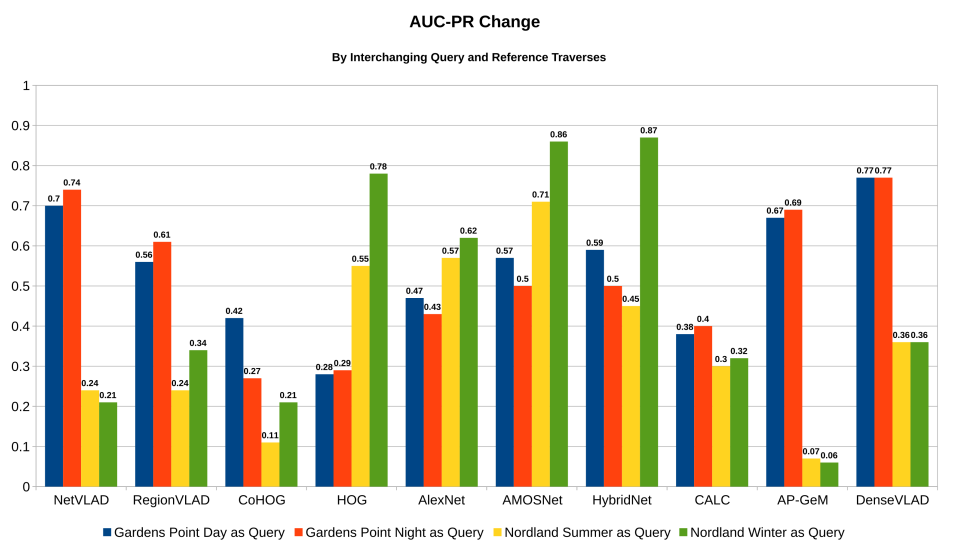


Figure 3.15: The effect on AUC-PR performance of techniques by inter-changing the query and reference traverses is shown here for the Gardens Point dataset and Nordland dataset.

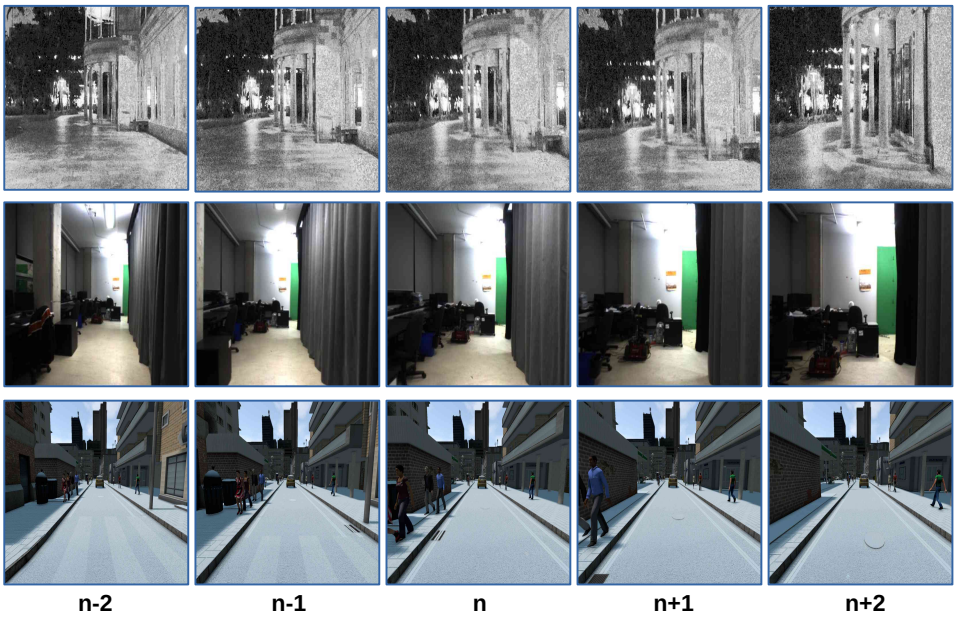


Figure 3.16: The overlap between visual information among subsequent images in traversal-based datasets is shown here. Depending on what level of ground-truth true positive range is acceptable, benefits will be distributed among the techniques based on their viewpoint-invariance.

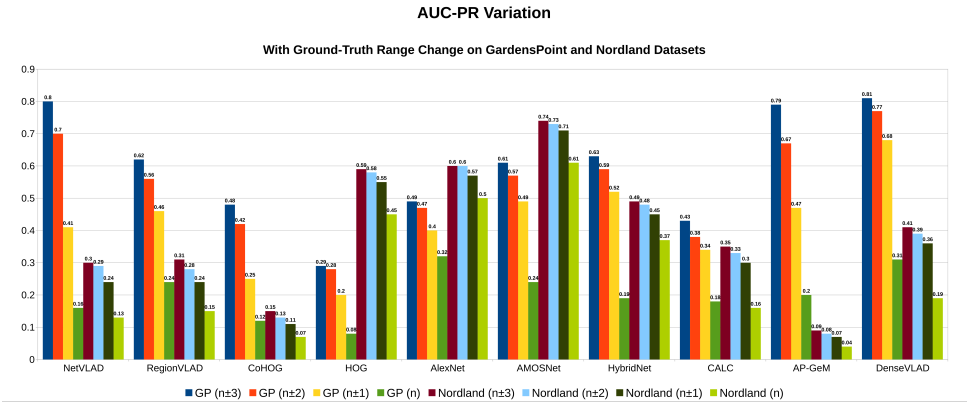


Figure 3.17: The effect on AUC-PR performance of techniques by changing the range of ground-truth true positive images is shown here for the Gardens Point dataset and Nordland dataset.

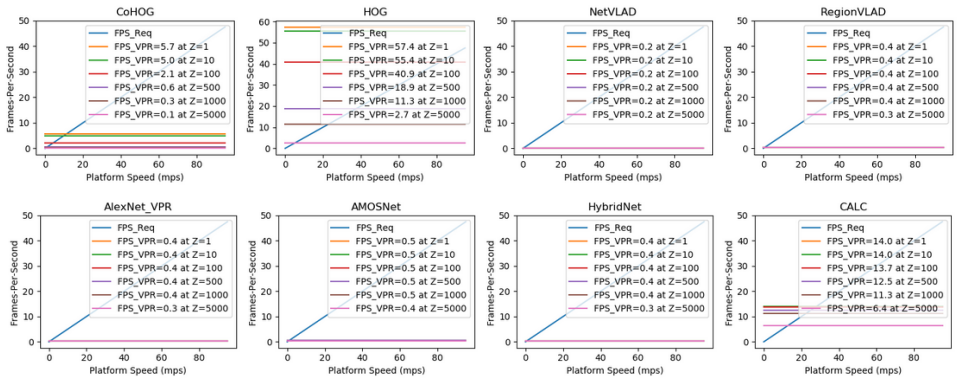


Figure 3.18: The retrieval performance of techniques is plotted for different map-sizes (Z) across the platform speed. Depending upon the value of frames required per meter (K) for an application, these curves will scale linearly according to equation 3.7.

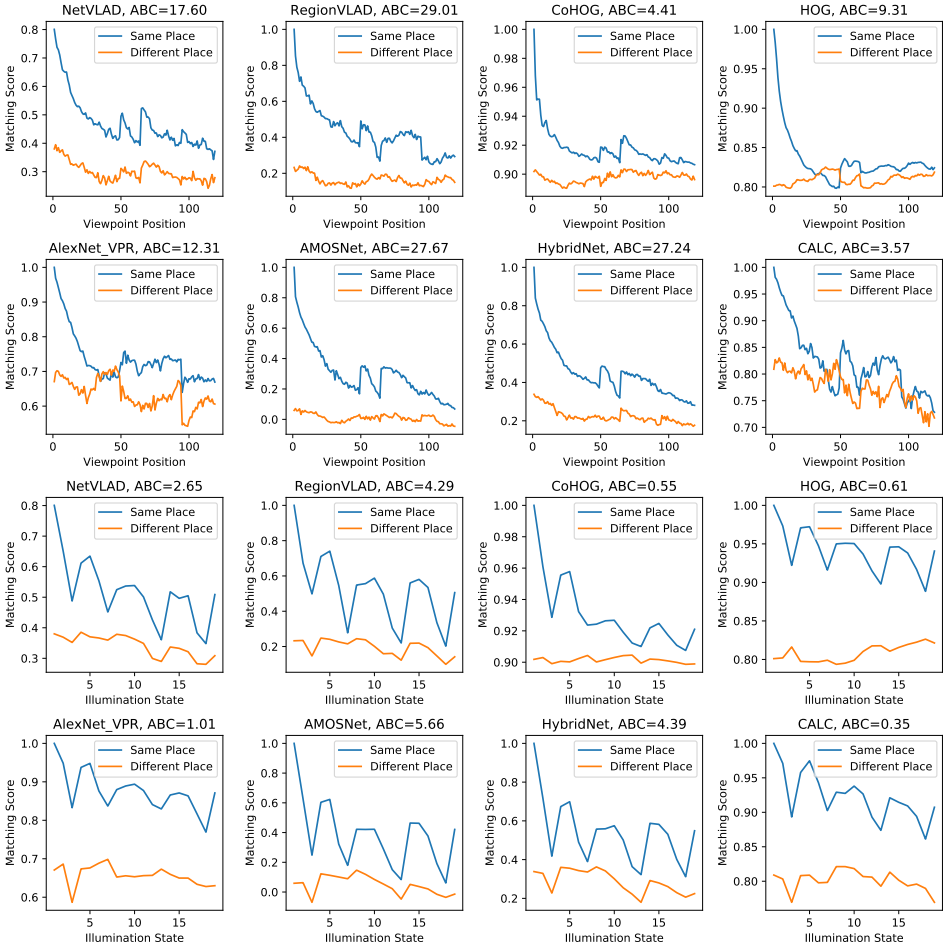


Figure 3.19: The change in matching score for quantified viewpoint and illumination variations is shown here on the Point Features dataset. The first two rows contain changes for all techniques with 119 viewpoint positions, while the bottom two rows show these changes for 19 different illumination levels. Please see the accompanying text for analysis.

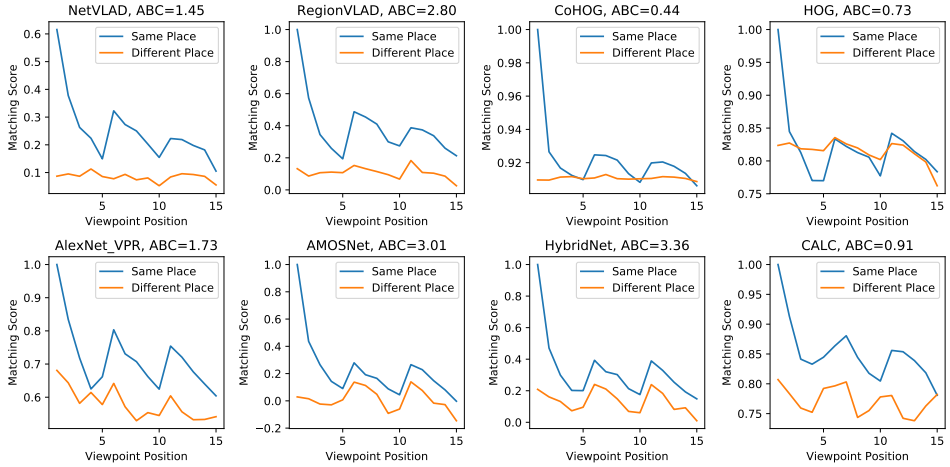


Figure 3.20: The change in matching score for the quantified viewpoint variations is shown here on the QUT Multi-lane dataset. The confidence score variation is shown for all techniques against the 15 viewpoint positions, as explained in sub-section 3.2.5.

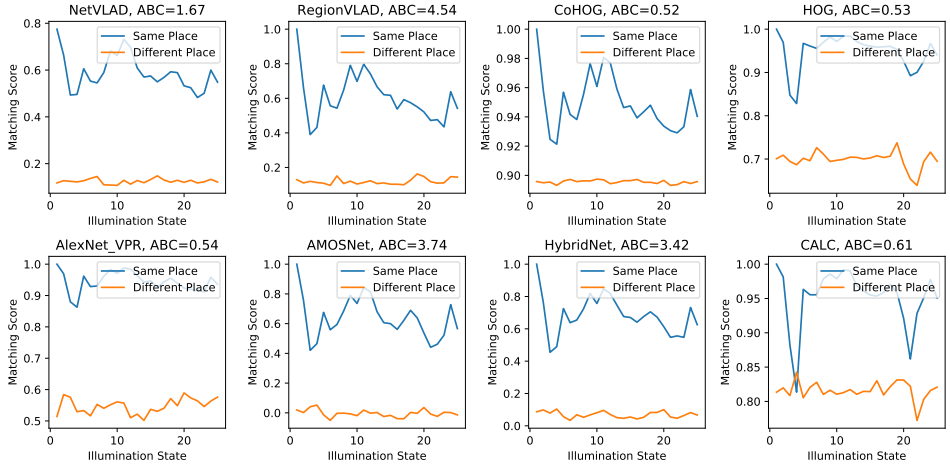


Figure 3.21: The change in matching score for the illumination variations is shown here on the MIT Multi-illumination dataset. The confidence score variation is given for all techniques on the 25 illumination positions, as explained in sub-section 3.2.5.

3.4 CONCLUSIONS OF THE CHAPTER

In this chapter, the author presented a comprehensive and variation-quantified evaluation framework for visual place recognition performance. The proposed open-source framework VPR-Bench integrates 12 different indoor and outdoor datasets, along with ten contemporary VPR techniques and popular evaluation metrics from both the computer vision and robotics communities to assess the performance of techniques on various fronts. The framework design is modular and permits future integration of datasets, techniques, and metrics in a convenient manner. The author utilized variation- and illumination-quantified Point Features dataset to evaluate and analyze the level and nature of variations that a VPR technique can handle. This analysis and the findings are extended from the synthetic Point Features dataset to the QUT Multi-lane dataset and the MIT multi-illumination dataset.

Using the proposed framework, a number of useful insights are provided about the nature of challenges that a particular technique can handle. It is identified that no universal state-of-the-art technique exists for place matching, and the reasons behind the success/failure of these techniques from one dataset to another are discussed. In the evaluations, DenseVLAD, a learning-based but non-deep-learning technique, has achieved state-of-the-art AUC-PR on 6 out of the 12 datasets, which indicates the potential for further developing the traditional specialized techniques and pipelines for VPR. It is also reported that 8 out of the 10 techniques have achieved state-of-the-art AUC-PR on at least one dataset, and therefore, ensemble-based approaches can present value towards creating a generic VPR system. The results reveal that the utility of VPR techniques highly depends on the employed evaluation metric, and that the corresponding utility is application-dependent, e.g., the state-of-the-art for RecallRate is different from that of AUC-PR because the former assumes the availability of a false-positive rejection scheme. Furthermore, the results demonstrate the utility of ROC curves for finding new places, which is usually not discussed in existing VPR literature. The encoding times for deep-learning-based techniques are significantly higher than handcrafted feature descriptors, but the availability of a GPU-based platform reduces this gap for most techniques. There are exceptions to this, e.g., RegionVLAD, a deep-learning-based technique that cannot benefit much from a GPU in terms of encoding time due to its CPU-bound intense region-extraction scheme. The author demonstrates that the descriptor matching time is dependent upon four factors: distance/similarity function, number of descriptor dimensions, length of each dimension, and the descriptor data-types. This identifies the need for further investigation of the trade-offs between reduced matching time at reduced descriptor precision and size. Overall, this chapter found that there is no one-for-all evaluation metric for VPR research, and that only a combination of these metrics presents the overall utility of a technique.

The results on the Point Features dataset identify that 3D viewpoint change has a more adverse effect on matching confidence than lateral viewpoint change, but deep-learning-based techniques generally suffer less from 3D change than handcrafted feature descriptors. It is further shown that directional illumination change presents a bigger challenge for VPR than uniform illumination change, both for deep-learning and handcrafted techniques. It is also proposed that viewpoint variance instead of viewpoint invariance can also be important for VPR systems, e.g., for accurate VPR-only localisation, sensitivity to viewpoint change can be a required feature. Because the author integrated a number of different datasets, techniques, and metrics, VPR-Bench enables many more performance comparisons, and

only a few selected comparisons have been discussed to limit the scope.

It remains future work to further investigate the relation between place matching performance and the bottlenecks caused by encoding times and linear scaling of matching times. The role of various parameters that determine the descriptor matching time is briefly introduced in this chapter, but it also deserves more detailed future investigation. It would also be useful to include evaluations on more challenging environments, such as underwater or aerial, on more extreme weather conditions, on motion-blur, and on opposing viewpoints. Further insights could be obtained by evaluating how different metrics yield different state-of-the-art VPR techniques on the same dataset.

Three key insights that were drawn from this chapter for the remainder of the thesis are: 1) there is no universal best VPR technique and train-test domain gap is a challenge, 2) good uncertainty estimation is needed in VPR for improved AUC-PR performance, 3) accurate localization using only VPR requires viewpoint-variant VPR methods and dense reference maps. Each of these insights is explored further as a dedicated chapter, respectively.

4

Using the Reference Map to Bridge the Domain Gap in VPR

This chapter is based on *M. Zaffar. The Overlooked Value of Test-time Reference Sets in Visual Place Recognition, ICCV, 2025.* [36].

Author contributions: Mubariz Zaffar proposed and implemented the method, performed the experiments, and took the lead in writing and presenting. Julian Kooij provided suggestions on the experimental design and technical writing. All other authors provided feedback on the writing and visualizations.

4.1 OVERVIEW

We know now that given a query image and a database of geo-tagged reference images, the task of a Visual Place Recognition (VPR) method is to retrieve from the database a correct matching reference image for this query. What is considered a correct match is ill-defined, but most VPR benchmarks consider any reference image within a fixed (e.g., 25-meter) circular radius of the query location as a correct match [9]. VPR has many applications, such as in landmark retrieval [129], 3D modeling [26], image search [181] and map-based localization [215, 216]. *These applications of VPR require that the test time reference set (the map) is available offline, i.e., before a test-time query is received.*[1]

Traditionally, the most investigated challenges in VPR have been viewpoint and appearance changes between the matching query and reference images, the so-called query-ref domain gap [5]. Thus, the objective of VPR methods is to extract representations robust to these variations. Given this objective, VPR benefited significantly through neural networks trained on large-scale VPR-specific datasets [9]. More recently, this has been complemented by adapting strong general-purpose *Vision-Foundation-Model backbones* (VFM) to the task of VPR, e.g., the DinoV2 vision transformer [6, 7, 87, 91]. As a result, test datasets with large query-ref domain gap (e.g., Tokyo-247 [80] and SVOX-night/snow [217]) that were previously challenging for VPR methods now seem solved ($\sim 98-99\%$ Recall@5) by the State-of-the-Art (SOTA) [6, 7, 91].

However, another important but less investigated challenge in VPR is the train-test domain gap, i.e., when the test dataset is from a different environment and/or device than the training dataset. It could be hypothesized that SOTA VPR methods would already be robust to this gap, since VFM backbones are known to generalize across datasets and tasks [218], and more so, when finetuned on diverse VPR-specific training data [35]. This hypothesis is examined in this chapter, revealing that the current SOTA VPR methods still suffer from the train-test domain gap. Details of this will follow later in the section 4.2.2.

To address the challenge posed by the large train-test domain gap in VPR, a strategy complementary to the typical curation of larger training datasets and/or using stronger VFM backbones is proposed. A case is made for using the unexplored reference set in test datasets to finetune the SOTA in VPR. It is argued that since this reference set with labeled (poses) images is freely available beforehand in various VPR applications and/or could even be obtained online, it is permissible to use it to bridge the train-test domain gap. Thus, outlining the two assumptions made in this chapter: a) the test-time reference set is available offline, b) there are resources available at test-time to finetune a VPR model.

Given this argument, the author illustrates in Fig. 4.1 the train-test domain gap in VPR. A T-SNE [219] projection of two VPR test datasets, Tokyo-247 [80] and Nordland [191], is shown along with the diverse GSV-Cities training dataset. The Tokyo-247 dataset contains urban scenes similar to the GSV-Cities and hence both form a single cluster, while the Nordland dataset contains railway tracks unlike GSV-Cities and forms a separate cluster. The author's proposal is simply that the reference set in test datasets (e.g., Nordland) could be combined with image augmentations to create a new finetuning dataset that has a smaller train-test domain gap than the original GSV-Cities dataset. Domain knowledge can then be

[1] The author acknowledges that there are other applications of VPR where the reference map may not be available offline, such as in SLAM. These applications are not the focus in this chapter.

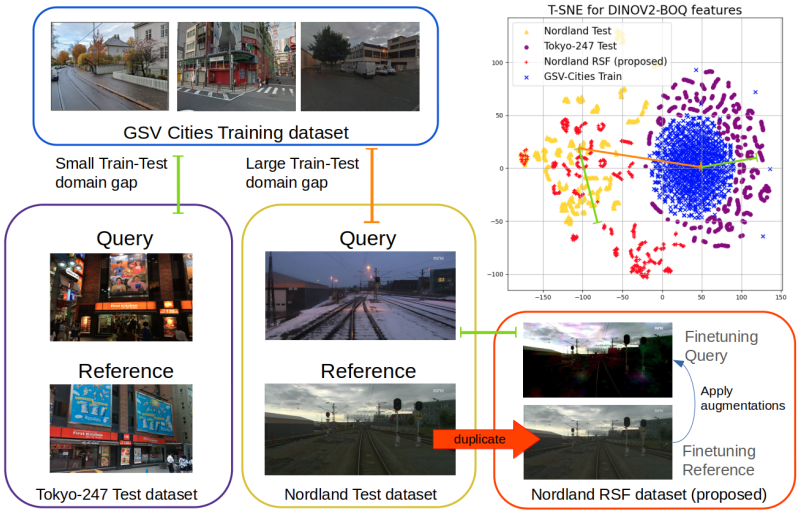


Figure 4.1: Large-scale VPR training datasets are usually created from Google Street View [35], e.g., the GSV-cities dataset. Thus, models trained in these environments perform well (SOTA Recall@5 $\sim 98-99\%$) for similar test datasets, e.g., the Tokyo-247 dataset [80], but suffer in unseen environments, e.g., the railway-tracks of the Nordland dataset [191]. A train-test domain gap exists, as evident in the T-SNE projection of descriptors computed using BoQ-DinoV2 [6] for randomly sampled images of these datasets. Descriptors from the Tokyo-247 dataset form a single cluster with the GSV-cities dataset, while the Nordland dataset is further away. Creating a finetuning dataset by using the freely available test-time reference images could help bridge the train-test domain gap.

injected into the model using this proposed finetuning dataset, akin to domain adaptation in other computer vision tasks such as classification [220].

However, this raises several questions: a) Is finetuning of VFM-based VPR methods on small test datasets useful? b) Do the finetuned models still generalize to other test datasets? c) Can a single finetuning strategy work across diverse test datasets? This chapter will present a simple self-supervised strategy, namely, Reference-Set-Finetuning (RSF), to answer these questions.

4.2 METHODOLOGY

In this section, VPR is quickly formalized in the context of this chapter, then the use of deep learning in VPR is formulated, and finally, the RSF strategy proposed in this work is described.

4.2.1 RECAPPING VPR FORMULATION

First, the author recaps the formulation of VPR for the reader. The goal of VPR is to find one or multiple reference images $I_i \in \mathcal{I}_\mathcal{R}$ that match the place of a query image $I_q \in \mathcal{I}_Q$ given a set of reference images $\mathcal{I}_\mathcal{R}$ with known poses $\mathcal{P}_\mathcal{R}$. The pose of I_q is then approximated by the pose of its nearest neighbour references in $\mathcal{I}_\mathcal{R}$. In its standard formulation, VPR consists of an offline map preparation stage and an online retrieval stage. The unknown pose p_q for the query I_q can then be approximated from the poses of the matched references

$p_i \in \mathcal{P}_\mathcal{R}$ [120].

In the offline phase, a VPR method G is applied to every reference image $I_i \in \mathcal{I}_\mathcal{R}$ to obtain D-dimensional reference feature descriptors $f_i = G(I_i)$. The method G is usually a trained neural network [196] or a handcrafted feature descriptor [197]. The resulting VPR map $\mathcal{M} = (\mathcal{I}_\mathcal{R}, \mathcal{R}, \mathcal{P}_\mathcal{R})$ contains the reference feature descriptors set $\mathcal{R} = \{f_1, \cdots f_N\}$, where each descriptor f_i is associated with a corresponding pose $p_i \in \mathcal{P}_\mathcal{R}$.

In the online retrieval stage, the same method G is applied to the query image I_q, and its descriptor $f_q = G(I_q)$ is compared to the reference descriptors in the map $\mathcal{M}$. This can be achieved through an efficient K-nearest neighbor lookup, considering the L2-distances $d_i = \|f_i - f_q\|_2$ between each reference i and the query q.

4.2.2 RELATING THE CURRENT SOTA IN VPR TO TRAIN-TEST DOMAIN GAP

VPR in deep learning is generally formulated either as a representation learning task [80] or a classification [221] task. The author uses the former formulation in this chapter. A deep-learning-based VPR method G consists of four major choices: a feature extraction backbone B, a feature aggregator P, a training dataset D, and a metric-learning loss function $\mathcal{L}$. The backbone B and aggregator P are compositional and together form the method G, such that $f_i = G(I_i) = P(B(I_i))$. This VPR method G is then trained on the training dataset D by minimizing the loss $\mathcal{L}$. The training dataset D is itself composed of four sets, such that $D = ({\mathcal{I}_\mathcal{Q}}^{train}, {\mathcal{P}_\mathcal{Q}}^{train}, {\mathcal{I}_\mathcal{R}}^{train}, {\mathcal{P}_\mathcal{R}}^{train})$, where for every $I_q \in \mathcal{I}_\mathcal{Q}$, the true and false matching reference images I_i are defined usually based on the spatial proximity of their corresponding poses in ${\mathcal{P}_\mathcal{Q}}^{train}$ and ${\mathcal{P}_\mathcal{R}}^{train}$, respectively, or based on visual overlap [222].

The choice of backbone in VPR is primarily motivated by advances in other vision tasks, and the community has seen a change from using VGG [80] and ResNet-based backbones [223, 224] to domain-agnostic Vision-Foundation-Model (VFM) backbones [6, 7, 90, 91]. For a given backbone B, different types of aggregators could be trained as P, for example, a NetVLAD layer [80], GeM layer [85], or the recently proposed Bag-of-learnable-Queries (BoQ) [6], etc. BoQ has been shown to outperform other aggregators trained on the same dataset with the same backbone [6].

Once the architecture $G = P(B(I_i))$ is fixed, the training loss $\mathcal{L}$ could be the distance-based loss [160], relative-pose-based loss [225], triplet loss [226], or the multi-similarity loss [227], etc. These losses could be minimized on different training datasets, for example, the Pitts-250k dataset [80], Mapillary Street Level Sequences dataset [128], San-francisco-XL [221] dataset, or the GSV-Cities dataset [35]. The purpose of these training datasets is to learn a generalizable feature extractor G that works well in different domains, and thus, the training datasets must be as diverse as possible. From existing literature, the GSV-cities dataset [35] is the most diverse training dataset in VPR.

Provided this formulation, would a VPR method G, employing a VFM backbone (e.g., DinoV2) trained on a large-scale diverse VPR dataset (e.g., GSV-Cities) with SOTA aggregation (e.g., BoQ), resolve the train-test domain gap? The author examines this by benchmarking the performance (Recall@5) in Table 4.1 of three DinoV2-based SOTA VPR methods that were published almost simultaneously [6, 7, 91]. All methods are trained on the GSV-cities dataset [35]: the most diverse training dataset in VPR, containing viewpoint and appearance changes from many streets across the world. The reported performance

	Backbone	SVOX-Snow	SVOX-Night	Pitts-250k	Tokyo-247	Nord.	Eyn.	Ams-AR	Avg.
Query-Ref gap		✓✓	✓✓	✓✓	✓✓✓	✓✓✓	✓	✓✓✓	
Train-Test gap		✓	✓	✓	✓	✓✓✓	✓✓	✓✓✓	
MixVPR [224] ('23)	ResNet50	98.4	79.5	98.2	91.7	86.8	93.2	60.4	88.5
BoQ [6] ('24)	ResNet50	99.5	94.7	98.5	95.9	91.1	94.9	75.4	93.8
Crica [7] ('24)	DinoV2	99.0	95.0	99.0	97.1	96.2	94.9	83.9	95.6
SALAD [91] ('24)	DinoV2	99.7	99.3	99.1	96.8	93.5	95.0	79.7	95.4
BoQ [6] ('24)	DinoV2	99.7	99.4	99.1	97.8	95.9	95.5	83.5	**96.4**

Table 4.1: *Recall@5* of some of the SOTA foundation-model-based VPR methods on various test datasets. All methods are trained on the most diverse VPR training dataset: the GSV-Cities dataset. The second row represents the domain gap of the respective test dataset from the GSV-Cities training dataset. ✓ indicates a small gap and ✓✓✓ indicates a large gap. On average, BoQ-DinoV2 is the SOTA in VPR, outlined in Bold, and thus the primary baseline. To indicate the margin of improvement left for BoQ, the datasets are ranked from left to right and colored. Datasets with a small train-test gap are almost solved, but a large train-test domain gap presents a challenge even for the SOTA VPR methods.

suggests that the test datasets with small train-test domain gap are almost solved by these SOTA VPR methods, despite their large query-ref domain gap. But some other test datasets, such as Nordland [191] and AmsterTime [228] with archival reference images, where the test environments differ significantly from the training dataset, still present a challenge.[2]

4.2.3 THE PROPOSED REFERENCE-SET-FINETUNING (RSF)

The preceding discussion suggests that although the training dataset D could be carefully curated to maximize diversity, it might still lack the domain knowledge needed for G to perform well on the test-time queries $\mathcal{I}_Q$. Here, the author makes his key observation: $\mathcal{I}_R$ is already available at the map preparation stage as well as its corresponding set of poses $\mathcal{P}_R$. Therefore, he proposes *Reference-set-finetuning (RSF)*, an unexplored but straightforward and effective procedure to adapt a trained model G to the target domain. Concretely, RSF (1) creates a **finetuning dataset** $D_{ft} = ({\mathcal{I}_Q}^{ft}, {\mathcal{P}_Q}^{ft}, {\mathcal{I}_R}^{ft}, {\mathcal{P}_R}^{ft})$, and (2) updates G on D_{ft} with pose-aware triplet mining, as illustrated in Fig. 4.2, and described in the following.

For D_{ft}, the finetuning query set ${\mathcal{I}_Q}^{ft}$ should represent a combination of viewpoint and appearance changes typically seen between the matching queries and references. Thus, a query $I_q^{ft} \in {\mathcal{I}_Q}^{ft}$ is formulated as $I_q^{ft} = A(I_i^{ft})$, where $A(.)$ represents an **augmentation operation**. Ideally, $A(.)$ approximates the viewpoint and appearance changes expected between the queries and references. An M number of different augmentations could be chosen as $A(.)$. In conclusion, the choices follow:

$$ {\mathcal{I}_R}^{ft} = \mathcal{I}_R, \tag{4.1} $$

$$ {\mathcal{P}_R}^{ft} = {\mathcal{P}_Q}^{ft} = \mathcal{P}_R, \tag{4.2} $$

$$ \text{and} \qquad |{\mathcal{I}_Q}^{ft}| = M \times |{\mathcal{I}_R}^{ft}|. \tag{4.3} $$

The finetuning queries ${\mathcal{I}_Q}^{ft}$ and references ${\mathcal{I}_R}^{ft}$ are encoded as feature vectors with G, positives and hard negatives [80] are **mined given the poses** ${\mathcal{P}_Q}^{ft}$ and ${\mathcal{P}_R}^{ft}$, and the

[2] Please note that I do *not* refer to the presence/absence of train-test domain gap in the various VPR test datasets in binary terms, but in a proportional manner. That is, while there is still a train-test domain gap between the GSV-cities dataset and the solved test datasets, this gap is larger for the unsolved datasets.

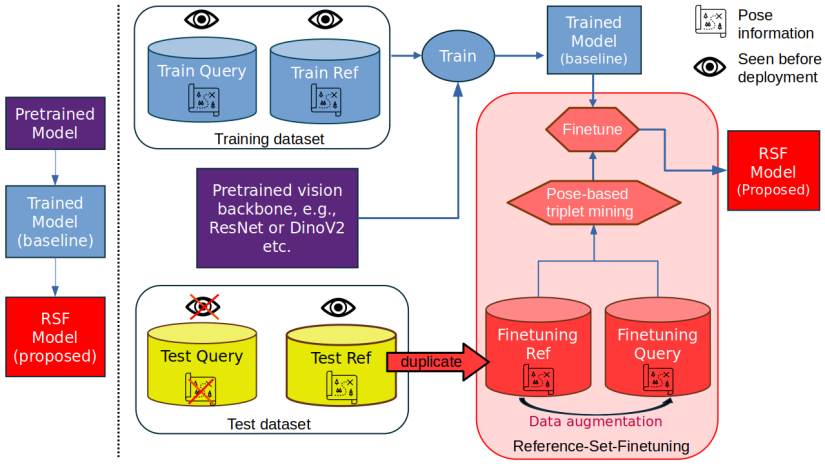


Figure 4.2: Deep learning for VPR usually utilizes a pretrained neural network that is further trained on a VPR dataset in a supervised manner with ground-truth poses. This usual pipeline assumes that we do not have any access to the test environment and that the training dataset is diverse enough to cover features of the test domain. However, there is always a train-test domain gap. The author proposes that the reference images in the test set are freely available offline in VPR and could be used to finetune VPR methods using simple data augmentations. This novel take on the problem setting of VPR, results in reference-set-finetuned (RSF) models that are more robust than the original trained model.

network G is **finetuned** using a standard triplet loss [94]:

$$L_{triplet} = max\{d(f_q^{ft}, f_p^{ft}) - d(f_q^{ft}, f_n^{ft}) + m, 0\}, \tag{4.4}$$

with a Euclidean distance function $d(f_1, f_2) = \|f_1 - f_2\|_2$ and a margin m. A hard-negative for a given query is the wrong reference image further than some fixed physical distance threshold that is the closest in the feature space.

4.3 EXPERIMENTS

First, the experimental setup of this chapter is presented, then the qualitative and quantitative performance of RSF models compared to baselines is reported, and finally the various aspects of RSF are evaluated.

4.3.1 DATASETS AND EVALUATION METRIC

To evaluate RSF, three public VPR datasets are used which have a large train-test domain gap and hence pose challenges to SOTA VPR methods, and one dataset with a small train-test domain gap. The ground-truth usage is similar to the standard formats in VPR [9], All of these datasets are summarized in Table 4.2.

The **Nordland dataset** [191] consists of a railway-track traversal through Norway during two different seasons: summer and winter. The summer traversal acts as reference images, while the winter images are queries. This dataset is challenging due to the unstructured environment depicted in different seasons. The challenging AmsterTime dataset [228] is also used which contains archival imagery of Amsterdam and its corresponding Google

	Queries	Refs.	Q-R gap	Train-test gap
Nord.	27.6k	27.6k	✓✓	✓✓✓
Amst-AR	1231	1231	✓✓✓	✓✓✓
Eyns.	24k	24k	✓	✓ ✓
SVOX-Ni	823	17.2k	✓✓	✓

Table 4.2: The datasets used in this chapter. The author reports the total number of query images, the total number of reference images, the presence of a domain gap between the queries and references, and the presence of a domain gap between the respective test dataset and the GSV-Cities training dataset. ✓ indicates a small gap and ✓✓✓ indicates a large gap.

Figure 4.3: Examples of the augmentations applied to create finetuning queries using Kornia augmentations [230]. Left-most is the original reference image.

Street View images. The author uses the archival images as references and street view images as queries, which depicts the task of retrieving an archival image of a place given a query image. This version is referred to as **AmsterTime-AR dataset**, outlining that the Archival images act as References. The **Eynsham dataset** [229] is used, which contains only grayscale images presenting a lack of color information for VPR. Finally, the **SVOX-Night dataset** [217] is used, which contains night-time images as queries and day-time images as references collected through Google Street View (GSV) in Oxford.

Following the existing literature, Recall@N is used as the evaluation metric. Ground-truths are as-is used by others [6, 7, 9, 91]. A retrieval is successful if the Top-N retrieved reference images were within a 25-meter radius of the query image.

4.3.2 IMPLEMENTATION DETAILS

Given the standards and SOTA described earlier in section 4.2.2, Dino-V2 [87] backbone with BoQ [6] aggregation trained on the GSV-cities dataset is used as the primary baseline VPR method G, since it is the current SOTA in VPR. Nevertheless, the author also reports the performance of SALAD [91] when used with the proposed RSF. The complete reference set of each respective test dataset is used for performing RSF as described in section 4.2.3. A small learning rate of 1e-7 is used for all datasets for both the VPR techniques. Simple image-level augmentations from the Kornia library [230] are used as A; examples are shown in Fig. 4.3. More sophisticated augmentations, such as domain translations using image-to-image vision foundation models, could also be considered [231]. The Kornia augmentations are applied on the fly and randomly chosen during training. To avoid overfitting the test set, the model is validated on the Pitts30k validation set [9]. RSF is done on a single NVIDIA A100 80GB GPU and, on average, takes only a few hours ($\approx 1-5$) depending on the size of the reference set and the method G.

	Nordland		Amster-AR		SVOX-Night		Eynsham		Average	
	R@1	*R@5*	*R@1*	*R@5*	*R@1*	*R@5*	*R@1*	*R@5*	*R@1*	*R@5*
MixVPR [224]	76.1	86.8	38.3	60.4	63.1	79.5	89.4	93.2	66.7	80.0
BoQ-Res [224]	83.3	91.1	52.1	75.4	85.7	94.7	91.2	94.9	78.1	89.0
CricaVPR [7]	91.2	96.2	64.7	83.9	86.9	95.0	91.6	94.9	83.6	92.5
SALAD [91]	85.9	93.5	58.7	79.7	95.0	99.3	91.5	95.0	82.8	91.9
BoQ [6]	90.4	95.9	61.9	83.5	97.1	99.4	92.1	**95.5**	85.4	93.6
SALAD-RSF	91.4	96.2	59.9	80.6	96.1	98.8	91.8	95.2	84.8	92.7
BoQ-RSF	**94.2**	**97.7**	**65.6**	**86.3**	**98.8**	**99.6**	**92.2**	95.4	**87.7**	**94.8**

Table 4.3: The recalls of SOTA VPR methods tested on various challenging test datasets. The first two rows: MixVPR and BoQ-Res use ResNet-50 backbone, while the remainder use DinoV2 backbone. All methods are trained on the GSV-Cities dataset. Best is in Bold.

4.3.3 RESULTS

Baseline comparison: Table 4.3 contains the performance of RSF models in comparison to baselines. Models finetuned using the proposed RSF outperform existing methods by a large margin for both metrics. Please note that this performance improvement is *without* the use of new training data or a stronger backbone. The performance benefits are more significant for the challenging Nordland and AmsterTime-AR datasets, which are the primary focus due to their large train-test domain gap. It is also noted that the proposed RSF is beneficial for the datasets without a large train-test domain gap, e.g., the SVOX-Night and Eynsham datasets. However, the performance improvement is less significant than on other datasets. More importantly, it is shown that both the SOTA VPR methods, BoQ and SALAD, benefit from RSF.

Fig. 4.4 further shows examples of queries that are correctly matched after the proposed RSF, and also some failure cases. Since BoQ with RSF is the best-performing method in the baseline comparison, the author focused on this method in the remainder of the experiments.

Model generalization: A key component of this study is the desire for the RSF models to retain generalization to the other test datasets. For this, the author reports in Table 4.4 the performance of an RSF model finetuned on a given reference dataset and evaluated on the other test datasets. Interestingly, it is noted that not only do the finetuned models retain generalization to other test datasets, but also that the RSF finetuned models consistently outperform the original model, agnostic to the reference set used for finetuning. This is attributed to the additional finetuning of SOTA on VPR-specific data; however, quite expectedly, we see a diagonal trend in the bold numbers, such that the best-performing RSF model for each test dataset is always the model finetuned on the same test dataset's reference map.

Attention masks: The attention masks for a learned BoQ query are visualized in Fig. 4.5 for the original model and the RSF model. Note that the RSF model strongly attends to the unique facades of windows in the building on the right, while the original BoQ only attends to edges.

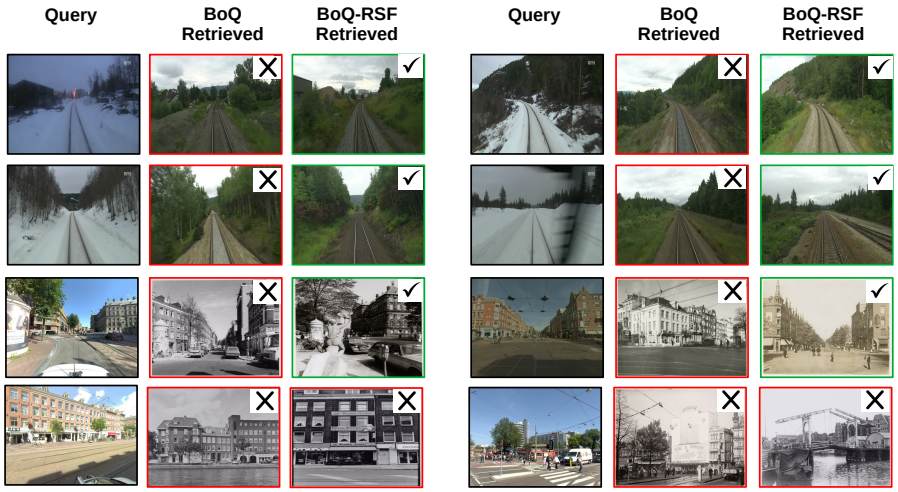


Figure 4.4: Examples of queries that are mismatched by the original BoQ-DinoV2 model but correctly matched by the reference-set-finetuned BoQ-RSF model, except for the last row, which demonstrates two BoQ-RSF failure cases.

	Test dataset		
	Nord.	Amst-AR	SVOX-Ni.
Baseline BoQ	90.4	61.9	97.1
BoQ-RSF (Nord.)	**94.2**	64.4	**98.9**
BoQ-RSF (Amst-AR)	92.3	**65.6**	**98.9**
BoQ-RSF (SVOX-Ni.)	93.4	64.7	**98.9**

Table 4.4: The Recall@1 of RSF models on various test datasets. The first column reports the reference set used for BoQ-RSF. RSF models retains generalization. Bold numbers in the diagonal indicate that the best-performing method for each dataset is the model finetuned on that dataset's reference set.

4.3.4 ABLATIONS

The author has argued in this chapter that the reference poses are freely available offline in VPR and are thus used in pose-based triplet mining for RSF. However, it is possible to have image-retrieval use-cases where reference images are available without pose information, e.g., image cataloging, landmark identification, etc. Table 4.5 thus reports the performance of the baseline in comparison to RSF models trained with and without access to pose information in the reference set. It is observed that although the reference pose information is helpful for RSF and such models are consistently the best-performing, but even without access to reference pose information, RSF models are still better than the baseline.

Table 4.6 shows the effect of Kornia augmentations on the proposed RSF for BoQ. These results show that augmentations are required to benefit from finetuning on the reference set, and that appearance augmentations are more useful than viewpoint augmentations for the chosen datasets. Only having viewpoint augmentations and no appearance augmentations is hurtful for RSF. The author hypothesizes that using viewpoint augmentations as A is

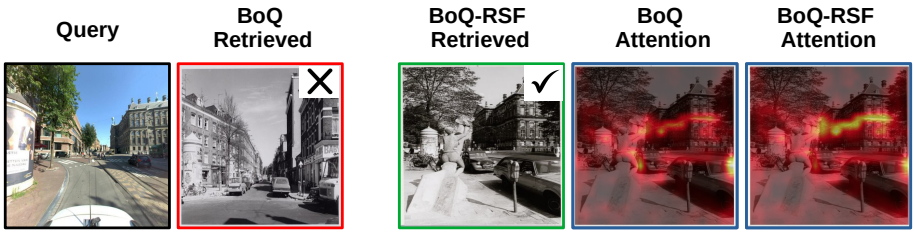


Figure 4.5: Learned attention for the original BoQ and the BoQ-RSF model on a ground-truth reference image is shown. The RSF model attends more to facades in the building, while BoQ attends to edges. These attention masks are for the *same* BoQ query of the original and the BoQ-RSF model.

	Nordland	Amst-AR
Baseline BoQ	95.9	83.5
BoQ-RSF (without poses)	97.1	85.3
BoQ-RSF (with poses)	**97.7**	**86.3**

Table 4.5: The Recall@5 performance of a baseline BoQ method is compared with RSF on two test datasets with and without access to the test-time reference poses. The availability of test-time reference poses allows for hard-negative mining and gives SOTA performance compared to random negative mining when pose information is not accessible. However, even without access to the reference poses, RSF model performs better than the baseline BoQ.

distracting for the model finetuned on the Nordland dataset, since there is almost no viewpoint change between the queries and the references in this dataset. The choice of augmentations in practice should follow from the expected query-reference domain gap, and in case of no prior knowledge about the expected Q-R gap, it is recommended that the viewpoint augmentations be used together with appearance augmentations as a thumb rule.

Chosen A	Amster-AR	Nordland
No augmentations	83.51	95.92
No viewpoint augmentations	<u>86.31</u>	**97.80**
No appearance augmentations	76.20	91.13
All augmentations	**86.32**	<u>97.70</u>

Table 4.6: The Recall@5 performance of BoQ-RSF with different types of augmentations chosen as A.

4.4 CONCLUSIONS OF THE CHAPTER

In this chapter, the author demonstrates that even the strong vision-foundation models-based VPR methods trained on large-scale Google Street View data struggle on test datasets that represent a domain different from the training data. It is thus argued that the reference set in test datasets is a free and valuable source of information that can be used to bridge this train-test domain gap. A simple Reference-Set-Finetuning (RSF) strategy is proposed that boosts the performance of SOTA VPR methods by large margins. The proposed RSF is shown to work for multiple datasets. The resulting finetuned models retain generalization to

other test datasets. The author also shows that the same RSF strategy could be applied to other VPR methods, albeit the performance benefits vary. Future works could investigate further how different formulations of RSF, particularly the augmentations, could benefit different VPR methods.

The author found that while the Reference-Set-Finetuning (RSF) performs consistently well for BoQ [6] across different test datasets, it does not perform well for all datasets for other techniques, such as CricaVPR [7] and SALAD [91]. Future works could explore alternative choices by looking at different kinds of augmentations, loss functions, pooling, and validation strategies that might suit these methods better. The author also believes that there is more room to explore other ways to bridge the train-test domain gap using the test-time reference set. A simple RSF strategy is proposed in this chapter, but advances in unsupervised domain adaptation could also be extended to VPR to fill this identified gap [232]. In addition, new benchmarks should contain this train-test gap as a representative challenge.

Having looked at the effect of the train-test domain gap, the next chapter is going to discuss how uncertainty in VPR can be estimated, especially since there will always be situations in which a VPR method may fail, and such a failure should be predicted.

5

Spatial-Uncertainty-Estimation (SUE) using the Test-time Reference Map in VPR

This chapter is based on *M. Zaffar. On the Estimation of Image-matching Uncertainty in Visual Place Recognition, CVPR, 2024.* [37].

Author contributions: Mubariz Zaffar proposed and implemented the method, performed the experiments, and took the lead in writing and presenting. Julian Kooij provided suggestions on the methodological formulation, experimental design, probabilistic formulation and technical writing. Liangliang Nan provided feedback on the writing and visualizations.

5.1 OVERVIEW

The previous chapter presented that even the state-of-the-art VPR methods trained on large-scale training datasets and using strong Vision-Foundation-Model backbones can fail in some cases. In this chapter, the author aims to predict such failures.

VPR is typically approached as an image retrieval problem, transforming images into feature vectors in a latent feature space where an efficient nearest neighbor search compares the query to all references. The pose of the query image is then approximated to be the same as that of the retrieved nearest neighbor references. Since successful VPR requires a good image representation that is robust to viewpoint and/or appearance changes [5, 196, 233], the field has benefited from advances in deep representation learning.

However, two images with similar visual content could still originate from geographically far-apart areas, a concept referred to as *perceptual-aliasing* in VPR [233]. For example, images with mostly sky could match many locations on an outdoor map. This constitutes aleatoric uncertainty, i.e., inherent noise or ambiguity in the data which cannot be reduced, as opposed to epistemic uncertainty which could be addressed with more training data [234]. The close proximity of perceptually aliased images in the feature space can result in catastrophic failures For instance, a highly confident false-positive from VPR could result in an incorrect loop closure in a SLAM pipeline, leading to misaligned maps [5, 25]. Reliable uncertainty estimation on the quality of the match is therefore key to avoid such failures by, e.g., rejecting results above a certain uncertainty threshold. Moreover, uncertainty estimation can also be used to fuse multiple predictions in VPR ensemble methods [118].

From existing literature, the author identifies three categories of methods to estimate image-matching uncertainty in VPR: retrieval-based uncertainty estimation (*RUE*), data-driven aleatoric uncertainty estimation (*DUE*), and geometric verification (*GV*) by local feature matching. **RUE:** Traditionally in VPR, the L2-distance between the query and the best-matched reference in the feature space has been used as an estimate of uncertainty [4]. The ratio of L2-distance between the first and second nearest neighbour reference is also used [116]. **DUE:** On the other hand, several recent works, such as the Bayesian Triplet Loss [117] and the Self-teaching Uncertainty Estimation [118], have proposed to explicitly learn to predict the aleatoric uncertainty from the query's image content only. **GV:** Another way to assert matching confidence is to test for consistent geometry among matched local features between the query and the best matching reference in a RANSAC loop [62].

Remarkably, none of the three categories exploits the spatial locations of matched images in the actual reference map, which the author hypothesizes can be an important source of information for estimating VPR matching uncertainty. To test this hypothesis, a new simple baseline is formulated, Spatial Uncertainty Estimation (SUE). SUE is a straightforward and efficient approach to estimating uncertainty for a query image's match, using the spatial spread of the physical poses for the most similar references in the map as a proxy. A high spatial spread indicates perceptual aliasing leading to high matching uncertainty, while a low spread indicates a distinct area is matched. An overview of the sources of information employed by all categories of methods and by SUE is provided in Table 5.1.

While all categories of uncertainty estimation methods aim for the same task, i.e., rejecting false positives in VPR, previous evaluations did not include all categories, providing an incomplete picture of the state-of-the-art. This chapter, therefore, compares the three existing categories and SUE on a levelled playing field, to provide recommendations for

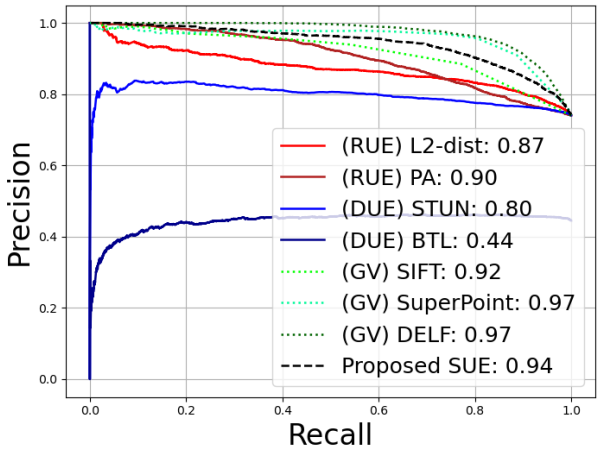


Figure 5.1: The Precision-Recall curves on the Pittsburgh dataset [80] for the three common categories of VPR uncertainty estimation methods (RUE, DUE, GV), and for the proposed baseline **SUE** which uniquely considers spatial locations of the top-K references. The global image descriptors [118] are fixed for all methods except BTL [117]. The *only* difference is the confidence given by each uncertainty estimation method to the best-matched reference descriptors for the corresponding queries. The legend lists the Area-under-the-Precision-Recall-curves. As *GV* methods are *two to three orders of magnitude* more computationally expensive than the others, they are plotted as dotted lines. Surprisingly, simple L2-distance in feature space is a better estimate of VPR uncertainty than recent deep learning-based uncertainty estimates. SUE outperforms all other efficient uncertainty estimation methods.

future research, and insights on the strengths/weaknesses of each category. For instance, as the preview of the experimental results in Fig. 5.1 indicates, SUE outperforms other efficient methods (this and other experiments will be discussed in more detail in section 5.3).

Concretely, the contributions of this chapter are:

1. A comparison of three different categories of uncertainty estimation methods in VPR.

2. A new simple baseline method, SUE, that considers the spatial locations of the reference images, a source of information not used by existing categories.

Categ.	**Descr.?**	**Poses?**	**Images?**	**Efficient?**
RUE	Top-K	No	No	Yes
DUE	No	Only train	Yes	Yes
GV	No	No	Yes	No
SUE	*Top-K*	*Top-K*	*No*	*Yes*

Table 5.1: Overview of the sources of information needed by the current main categories for VPR uncertainty estimation, and by the proposed method *SUE*: the query/reference global image descriptors, the reference poses, or complete query/reference images. Efficiency refers to the inference time needed by each approach.

3. Since *GV* gives the best uncertainty estimates albeit at a higher computational cost, the author investigates whether the other methods are complementary to *GV*.

5.2 METHODOLOGY

This section first introduces the task of uncertainty estimation for VPR. The formulation of VPR is then recapped, and the three main categories of uncertainty estimation methods are described. Next, the author formulates the proposed baseline approach, SUE, which unlike the other three categories uses the freely available reference poses information. Finally, the author outlines how he combines the different categories of methods with the computationally expensive geometric verification to investigate if the uncertainty estimates are complementary.

5.2.1 UNCERTAINTY ESTIMATION IN VPR

Typically VPR is considered as an image retrieval task: finding the most similar reference images to the query by Euclidean distance in some feature space. The poses associated with the images, however, distinguish VPR from other image retrieval tasks, such as web search, where matches are correct if their image content is judged as the same. In VPR, we often instead refer to the location of the query and references to judge matches: a retrieved reference is only acceptable if its pose is within a maximum distance threshold of the (unknown) true pose of the query [8, 9, 233]. Ideally, the closest matches in the feature space thus also have the poses closest to the query pose. However, this is often not the case in VPR due to *perceptual aliasing*, a form of aleatoric uncertainty since it cannot be reduced by choosing a different feature encoder or by using more training data.

It is therefore desirable to obtain some *uncertainty score* s_q for a query and the retrieved nearest neighbor, where a low score expresses confidence that the nearest neighbor is a correct match. A threshold τ on the score could then reject a query ($s_q > \tau$) for which the best match is at risk of being incorrect to prevent failures of the downstream application [233]. The objective of VPR uncertainty estimation is thus to score queries, such that queries with reliable matches can be distinguished from those with possible incorrect matches. Note that while an uncertainty estimation method could provide scores with an explicit probabilistic interpretation, this is not a strict requirement to apply an acceptance threshold.

5.2.2 RECAPPING VPR FORMULATION

Given a set of reference images $\mathcal{I}$ with known poses $\mathcal{P}$, the goal of VPR is to find one or multiple reference images $I_i \in \mathcal{I}$ that match the place of a query image I_q. The unknown pose p_q for the query I_q can then be approximated from the poses of the matched references $p_i \in \mathcal{P}$, since correct matches should have been taken in the same area. The exact formulation of a pose generally depends on the localization source and the task, for example, 2D GPS coordinates for visual geo-localization [9], or 6D pose [21]. In this research, the author follows a general task-independent formulation and only assumes that a pose p_i consists of 2D or 3D spatial coordinates in some global coordinate system.

In the offline map preparation phase of VPR, before accepting queries, a feature extractor G is applied to every reference image $I_i \in \mathcal{I}$ to obtain D-dimensional reference feature descriptors $f_i = G(I_i)$. Usually G is a trained neural network [196] or a handcrafted feature

descriptor [197]. The resulting VPR map $\mathcal{M} = (\mathcal{R}, \mathcal{P})$ contains the reference feature descriptors set $\mathcal{R} = \{f_1, \cdots f_N\}$, where each descriptor f_i is associated with a corresponding pose $p_i \in \mathcal{P}$.

At test time, the same feature extractor G is applied to the query image I_q, and its query descriptor $f_q = G(I_q)$ is compared to the reference descriptors in the map $\mathcal{M}$. This can be achieved through an efficient K-nearest neighbor lookup, considering the L2-distances $d_i = \|f_i - f_q\|_2$ between each reference i and the query. This gives an ordered list of K nearest neighbor references $\mathcal{R}_{\text{nn}} = [f_{(1)}, \cdots, f_{(K)}]$, ranked by increasing distance $d_{(1)} \leq \cdots \leq d_{(K)}$ and with corresponding poses $\mathcal{P}_{\text{nn}} = [p_{(1)}, \cdots, p_{(K)}]$. Here the author uses bracketed subscript (j) to indicate j-th item in the ranked order, i.e., $f_{(1)} = \text{argmin}_{f_i \in \mathcal{R}} \|f_i - f_q\|_2$ is the descriptor with the smallest distance to the query in the feature space.

Each corresponding pose $p_{(i)} \in \mathcal{P}$ can be considered as a hypothesis to estimate the query's true pose p_q, though usually only the pose of the best matching reference feature descriptor $f_{(1)}$ is considered as the VPR pose estimate p'_q for the query, i.e., $p'_q = p_{(1)}$ [8]. This best-match-based query pose estimation is followed in this work. In benchmarks, a match is considered correct if p'_q is 'physically near' to p_q. The threshold on what distance is still accepted as the same 'place' depends on the scale of each localization task [5].

5.2.3 CURRENT VPR UNCERTAINTY ESTIMATION CATEGORIES

The various representative uncertainty estimation methods are now described for the three common categories.

Retrieval-based uncertainty estimation (*RUE*): Commonly, the matching uncertainty in VPR is considered proportional to the L2-distance from the best match $d_{(1)}$, so $s_q = d_{(1)}$ [8, 9], as this distance indicates relevant differences between the visual content of the query and match.

An alternative is to consider the distance ratio between the first and second nearest neighbor, $s_q = d_{(1)}/d_{(2)}$. This ratio is quite similar to the perceptual aliasing score (PA score) [116] and the false-positive rejection criterion in the popular descriptor SIFT [39].

Data-driven uncertainty estimation (*DUE*): State-of-the-art VPR encoders are typically deep neural networks trained on a labeled VPR dataset. The labeled training data contains the ground-truth poses $\mathcal{P}_{\text{train}}$ for the training references and query images $\mathcal{I}_{\text{train}}$. A deep encoder G can be adapted to also predict the aleatoric uncertainty of matching a nearby pose, by learning from the training query image in $\mathcal{I}_{\text{train}}$ when an image is distinctive and obtains good pose matches within $\mathcal{P}_{\text{train}}$, and when not (e.g., images of trees, uniform walls, or sky). Methods in this category include the Bayesian Triplet Loss (BTL) [117], and STUN [118]. Note that the learned uncertainty is based on the training images and poses, not those in the test-time reference map $\mathcal{M}$.

In general, an uncertainty-aware encoder $(\bar{f}_i, \sigma_i^2) = G'(I_i)$ predicts for an image I_i not only the expected feature $\bar{f}_i$, but also the *variance in the feature space*, i.e., $f_i \sim \text{N}(\bar{f}_i, \sigma_i^2)$. The total variance in σ_i^2 can be used as a proxy for the image-matching uncertainty, $s_q = \|\sigma_i^2\|_1$. The computational overhead of the deep network producing an additional output σ_i^2 is low.

Geometric verification (*GV*): Another way to estimate image-matching uncertainty is to compare the query and the best-matched reference image in more detail through local feature matching and geometric verification in a RANSAC loop, e.g., through the use of SIFT [39], DELF [62], and SuperPoint [60]. All the matched local features that satisfy a geometric

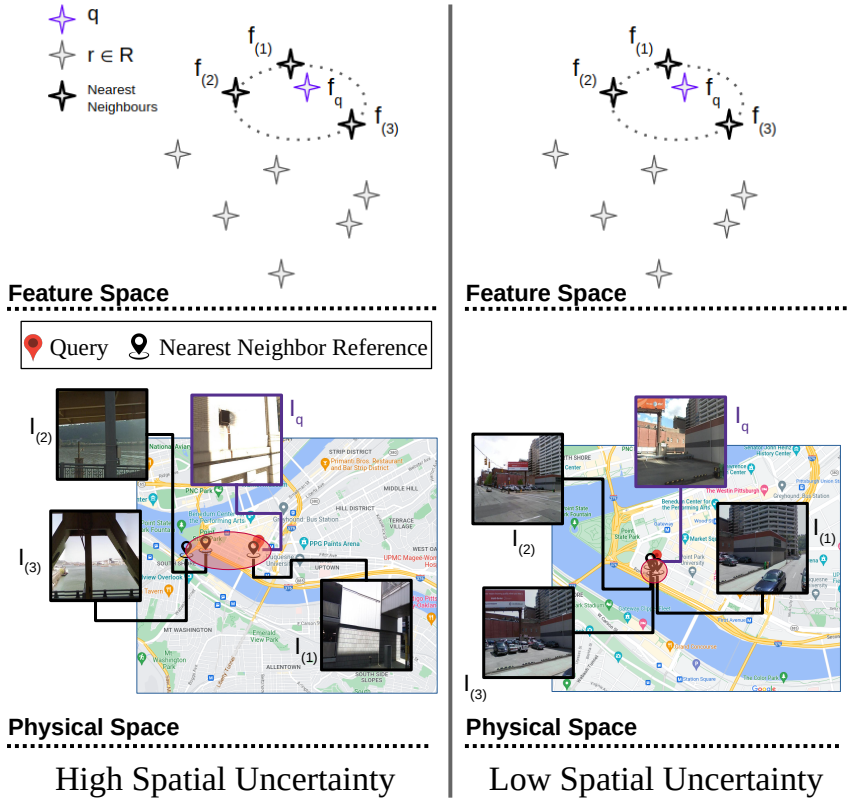


Figure 5.2: In VPR, a query q is compared in feature space to features $f_i \in \mathcal{R}$ of reference images with known poses. The nearest neighbors $f_{(1)}, \cdots, f_{(K)}$ are retrieved as matches. Left: The retrieved references $I_{(1)}, I_{(2)}, I_{(3)}$ share similar visual content with the query (walls, pillars, and blobs), but are geographically far apart, reflecting high uncertainty that the matched reference is correct. Right: For another query, the retrieved references are geographically close together, indicating low uncertainty.

transformation estimated from the randomly sampled set of matched local features between the query image and the reference image are considered inliers. The confidence is indicated by the number of inliers c_{gv}, which could be expressed as a matching uncertainty estimate, i.e., $s_q = -c_{gv}$. While geometric verification yields high-quality uncertainty estimates, such post-processing is computationally expensive compared to the other methods.

5.2.4 SPATIAL UNCERTAINTY ESTIMATION (*SUE*) FOR VPR

The author observes that the poses in the reference set $\mathcal{P}$ are a potentially powerful and freely available source of information at *test time*, which current uncertainty estimation methods do not exploit (more details will be presented in Sec. 5.2.1). The intuition behind this is illustrated in Fig. 5.2, where the author shows that if the nearest neighbors in the feature space are spatially far apart in their respective 2D/3D world coordinates, it indicates that such a feature suffers from perceptual aliasing: various areas in the test reference set contain the queried appearance, thus, uncertainty on the pose estimate should be high. On the other hand, if the nearest neighbors in the feature space are also spatially close together, there is agreement among the matching pose hypotheses that the matched area is distinct within that given reference set, thus, the uncertainty should be low.

To test this insight, the author now formulates SUE, a purposefully simple image-matching uncertainty estimation method. Given the K-best retrieved references, fit a 2D or 3D multivariate Gaussian distribution $\mathrm{N}(\mu_p, \Sigma_p)$ over their 2D or 3D poses $\mathcal{P}_{\mathrm{nn}}$,

$$\mu_p = \frac{1}{\sum_i w_{(i)}} \sum_{i=1}^{K} w_{(i)} \cdot p_{(i)}, \tag{5.1}$$

$$\Sigma_p = \frac{1}{\sum_i w_{(i)}} \sum_{i=1}^{K} w_{(i)} \cdot (p_{(i)} - \mu_p)(p_{(i)} - \mu_p)^\top, \tag{5.2}$$

where the relative contribution $w_{(i)}$ of the i-th best reference pose $p_{(i)}$ decreases as its L2-distance $d_{(i)}$ to the query in the feature space increases,

$$w_{(i)} = e^{-\lambda \cdot d_{(i)}}, \quad \text{where} \quad d_{(i)} = \|f_q - f_{(i)}\|_2. \tag{5.3}$$

The total variance across the spatial pose dimensions could then serve as a proxy for image-matching uncertainty, i.e., $s_q = \mathrm{trace}(\Sigma_p)$.

The hyper-parameter λ controls the non-linear relative contribution of a pose p_i for the nearest neighbor $f_{(i)} \in \mathcal{R}_{\mathrm{nn}}$ given its distance $d_{(i)}$ in the feature space. This hyper-parameter can be optimized on training data, though the later experiments will show that its choice is remarkably robust across various real-world benchmark datasets.

5.2.5 COMPLEMENTING GEOMETRIC VERIFICATION

To study to what extent SUE's (or another method's) s_q provides information not captured by the c_{gv} metric from geometric verification, the author treats both scores as a 2D feature vector and trains a classifier to predict if a best-matched reference should be accepted as a true-positive, or rejected as a false-positive. The regular rejection threshold is extended from a single score ($s_q > \tau$) to a linear weighted sum of both scores ($s_q/\tau_1 + c_{gv}/\tau_2 > 1$), by the use of a regular linear Support Vector Machine (SVM) as a classifier.

Method	↑ Pitts.	↑ Sanfr.	↑ Stluc.	↑ Eyn.	↑ MSLS	↑ Nordland	↑ Average	↓ Time
(RUE) L2-distance	0.87	0.76	0.79	0.87	0.64	0.18	0.69	**0.05**
(RUE) PA-Score [116]	0.90	0.65	0.77	0.88	0.68	0.21	0.68	**0.05**
(DUE) BTL [117]	0.44	0.17	0.34	0.45	0.21	0.07	0.28	0.20
(DUE) STUN [118]	0.79	0.57	0.66	0.71	0.44	0.05	0.54	0.10
SUE	**0.94**	**0.84**	**0.88**	**0.93**	**0.77**	**0.26**	**0.77**	1.08
(GV) SIFT-RANSAC [39]	0.92	0.89	0.93	**0.96**	0.70	0.15	0.76	**129**
(GV) DELF-RANSAC [62]	**0.97**	0.92	**0.97**	0.95	**0.95**	**0.84**	**0.93**	1587
(GV) Super-RANSAC [60]	0.95	**0.95**	**0.97**	**0.96**	0.87	0.50	0.87	848

Table 5.2: The AUC-PR of all the compared methods. Higher AUC-PR is better, and best is in Bold. The bottom rows are the computationally expensive geometric verification methods. The last column lists the time (msec) to give an uncertainty estimate for a single query image.

5.3 EXPERIMENTS

First the setup for experiments is presented. Then, the performance of all the image-matching uncertainty estimation methods is compared on multiple benchmark datasets. Next, the author tests if the methods are complementary to geometric verification. Finally, an ablation over the hyper-parameters of SUE is performed, and a discussion is provided.

5.3.1 EXPERIMENTAL SETUP

This section describes the datasets, baselines, evaluation metrics, and implementation details of this chapter.

Datasets: The author uses six public VPR datasets in this chapter: Pittsburgh-250k [80], Sanfrancisco [74, 206], Stlucia [235], Eysham [229], MSLS [128], and Nordland [191]. Details of these datasets and their respective ground-truths is provided in Berton et al. [223].

Baselines: The primary baselines for uncertainty estimation include the L2-distance in feature space d_q, the perceptual aliasing score (PA score [116]), the Bayesian Triplet Loss (BTL) [117] and STUN [118]. As the code for BTL is not open-source, the author implements it following the pseudo-code and the network details provided in the original paper.

For geometric verification, the author tests three types of local feature descriptors, namely the handcrafted SIFT [39], the deep-learning-based DELF [62], and SuperPoint [60], which are referred to as SIFT-RANSAC, DELF-RANSAC, and Superpoint-RANSAC, respectively.

Evaluation metrics: The precision-recall (PR) curves have been widely used in VPR for estimating the retrieval quality [8] and have also been motivated in Chapter 3. However, they can also be used to evaluate the quality of uncertainty estimation in VPR, as widely used in existing uncertainty estimation tasks in deep learning [236, 237]. The choice of PR-curves over the Receiver Operating Characteristic (ROC) curve is due to the absence of true-negatives in employed VPR datasets. Given a fixed list of retrieved images, the Precision-Recall curves can reflect the technique with the better uncertainty estimates s_q. A technique that can perfectly classify between true-positives (TP) and false-positives (FP), given the uncertainty estimates, achieves an Area-under-the-Precision-Recall-Curve (AUC-PR) of 1.

For the combination of uncertainty estimates with geometric verification, the task is formulated as binary classification, and accuracy is used as an evaluation metric based on

the ground-truth true-positives and false-positives [223].

Implementation details: For SUE and all the other baselines except BTL, a ResNet-50 backbone is used with GeM pooling, trained in a self-teaching manner as in Cai et al. [118] on the training split of the Pittsburgh dataset. Each feature vector f_i is 2048-dimensional. For BTL, the same backbone and training data are used, but using the training procedure specified in the original BTL paper [117]. For DELF and SuperPoint, the implementations are open-sourced by the respective authors, and the default settings are employed. For SIFT-RANSAC the author uses the OpenCV implementation with the number of extracted features set to 5000, the Lowe test ratio to 0.6, and the number of RANSAC iterations to 1000.

The hyperparameters in SUE are fine-tuned only on the Pittsburgh dataset and then fixed as $\lambda = 350$ and $K = 10$ for all datasets and experiments. An ablation over these parameters is given later in section 5.3.4. The SVM is trained with stochastic gradient descent with hinge loss and an L1-penalty, and a maximum of 1000 training iterations.

5.3.2 PERFORMANCE COMPARISON

Firstly, all the uncertainty estimation methods formulated in this chapter are compared, both qualitatively and quantitatively, and in terms of their computational overhead.

Area-under-the-Precision-Recall-curves: The AUC-PR for all the methods on all the datasets are summarized in Table 5.2. SUE outperforms other efficient methods by a clear margin, even on the Pittsburgh dataset which was used for training STUN and BTL. It is also important to note that a basic L2-distance-based uncertainty already outperforms BTL and STUN. Moreover, geometric verification outperforms all other uncertainty estimates although SUE achieves comparable performance.

Computational requirements: The author further reports the time taken to compute the *GV* confidence s_q and the uncertainty estimates in Table 5.2. Although *GV* gives useful uncertainty estimates, the high computational cost of these *GV* methods may be prohibitive for real-time online applications. SUE is about *three orders of magnitude* faster than *GV* using DELF-RANSAC.

Qualitative results: To obtain insight into how the different methods interpret the visual content in query images and what they are sensitive to, Fig. 5.3 shows examples of the most and the least uncertain query images for different methods in the Pittsburgh dataset. While all methods usually consider feature-rich and distinctive buildings as least uncertain for VPR, differences between the methods lie in the most uncertain images. Highly saturated test images are considered most uncertain by L2-distance-based uncertainty because such saturation did not exist during the reference traversals of the same scene. On the other hand, STUN considers images of trees and walls that usually contribute to perceptual aliasing as the most uncertain for VPR. SUE considers traffic squares and common building patterns as the most uncertain. Note that because SUE uses the freely available pose information in the test reference set; whether a traffic square or a building is considered uncertain is specific to this test reference set and not due to a generally-applicable visual property.

Fig. 5.4 further shows several images that illustrate failure cases of *RUE* and *DUE* in comparison to *SUE*. Images of walls generally contribute to high aleatoric uncertainty (*DUE*) and are closer together in the feature space in terms of L2-distance (*RUE*). However, note that the query in Fig. 5.4 Top is correctly matched since only a unique wall with this pattern exists

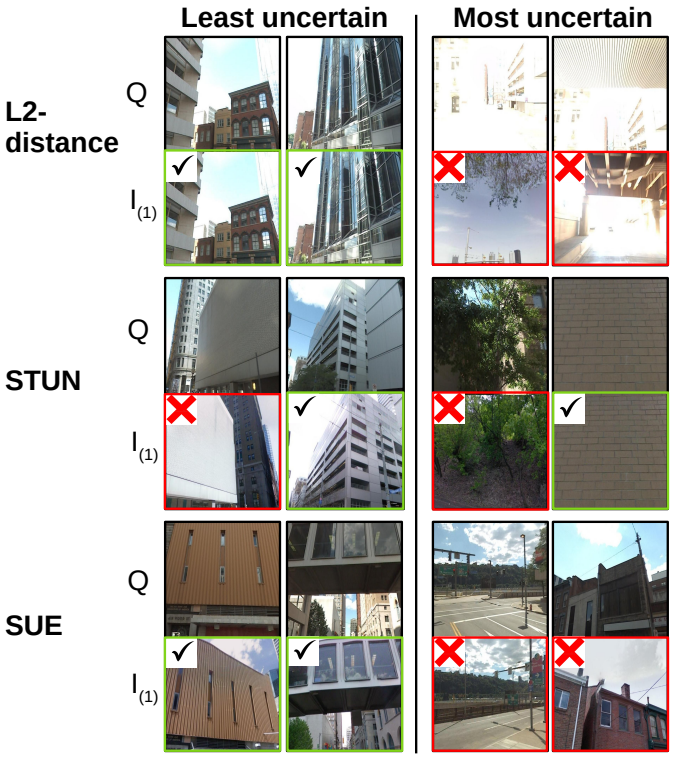


Figure 5.3: Examples of the two least and the two most uncertain query images with the corresponding nearest neighbor on the Pittsburgh dataset. The colors/symbols indicate whether the retrieved image is a correct match.

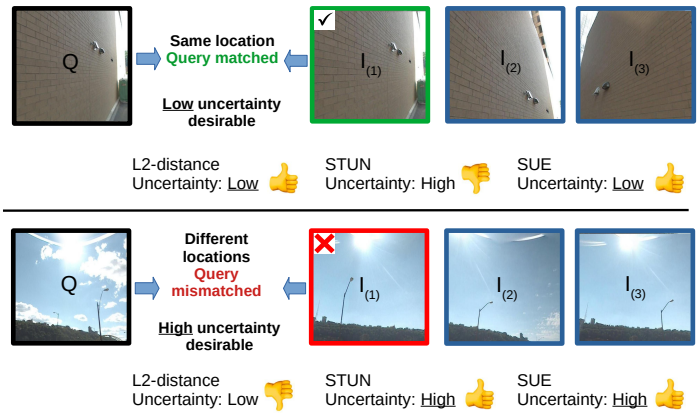
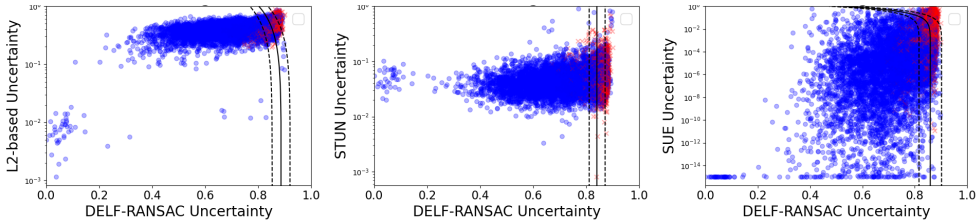


Figure 5.4: Two queries and their nearest neighbor reference images that illustrate cases where SUE outperforms other methods. Ideally a method assigns high uncertainty to the mismatched query and low uncertainty to the correct match, as SUE does here.



(a) (b) (c)

Figure 5.5: The relation between geometric verification uncertainty (x-axis) and the L2/STUN/SUE uncertainty (y-axis) on the Pittsburgh dataset [80]. Each point represents a query, with *blue* indicating a correct match, and *red* otherwise. The linear SVM boundaries are shown as black lines, while the dashed lines are the SVM margins. Scores have been linearly scaled to the $[0,1]$ range based on the min/max value in the training data, and for better visualization the vertical scale is in log-space, hence the SVM boundaries appear non-linear. The class distributions in the right-most plot reveal that SUE complements geometric-verification, especially when the latter has low confidence.

Figure 5.6: Correctly matched queries that are given high uncertainty by DELF-RANSAC and low uncertainty by the proposed method SUE.

in the test reference set. SUE and L2-distance correctly give this query a low uncertainty, but STUN fails. The query image in Fig. 5.4 Bottom is given low uncertainty by the L2-distance than ranking with STUN and SUE. This is because images with large portions of sky contribute to aleatoric uncertainty, but they are close in terms of the feature space L2-distance. This query is mismatched and identifies where L2-distance-based uncertainty fails in comparison to STUN and SUE.

5.3.3 COMPLEMENTING GEOMETRIC VERIFICATION

Finally, the author tests whether efficient uncertainty estimation can complement geometric verification, as outlined in Sec. 5.2.5. Fig. 5.5 shows the relation between the different types of uncertainties with the uncertainty from geometric verification. As we note from Table 5.2, STUN outperforms BTL, and L2-distance is on average better than the PA-score, thus the author only combines STUN, L2-distance, and SUE with geometric-verification here.

SUE provides complementary performance by giving low uncertainty s_q to images that are correctly matched but which were given high *GV* uncertainty. Some of these

Method	Pitts.	San.	Stlu.	Eyn.	MSLS
Superpoint	85.1	53.2	76.4	67.5	36.9
DELF	86.0	86.6	85.3	78.3	**80.2**
L2-distance	75.7	57.3	56.2	67.7	36.8
STUN	74.0	54.0	58.0	67.6	37.4
SUE	78.9	70.7	72.8	77.3	46.0
DELF+L2-di.	85.7	86.1	82.3	77.3	72.0
DELF+STUN	85.4	81.6	80.1	75.0	68.2
DELF+SUE	**87.1**	**89.6**	**88.7**	**82.1**	**73.4**

Table 5.3: Binary classification accuracy given the uncertainty estimates of various methods, using a linear SVM trained *only* on the Pittsburgh dataset. The combination *DELF + SUE* generalizes better than baseline combinations, except on the MSLS dataset where although *DELF+SUE* is better than the other combinations, the SVM boundaries learned from Pittsburgh are not the best.

complementary queries are shown in Fig. 5.6, where it can be seen that these queries are images that are generally difficult to match local feature descriptors, such as facades, trees, and other repetitive features within the image [238]. The linear boundaries learned by SVM to classify between true-positives and false-positives are also shown. The classification accuracy of the different methods is reported in Table 5.3.

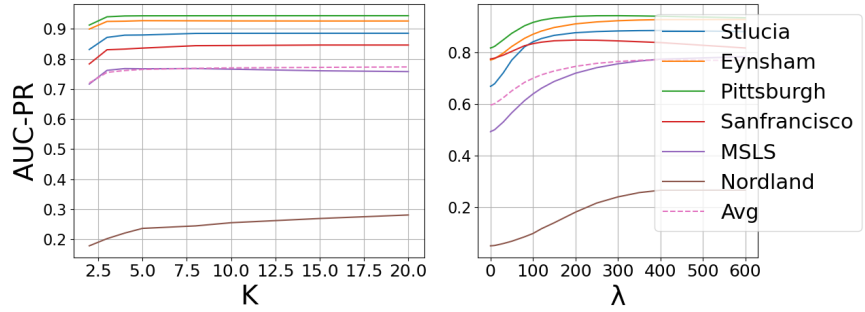
5.3.4 ABLATION STUDY

SUE requires two hyper-parameters, the number of nearest neighbors K and the decay parameter λ that controls the relative contribution of the poses of the nearest neighbors. The author shows the ablation over these parameters in Fig. 5.7 by plotting the corresponding AUC-PR values for all datasets given a set of values for each parameter. The trend remains primarily the same across all datasets. Note that the AUC-PR increases by considering more nearest neighbors, but the curves mostly plateau after $K = 5$, since poses from low-ranked neighbors contribute less to the overall pose hypothesis. For λ, we can see that the range $200 - 400$ is generally stable and gives reliable uncertainty estimates.

The author performs two further experiments: changing the backbone feature extractor from STUN [118] to CosPlace [221] to show SUE's generality to other backbones in Fig. 5.8, and the benefit of using the exponential weighing function (in Equation (5.2)) instead of the uniform weighing, as reported in Table 5.4.

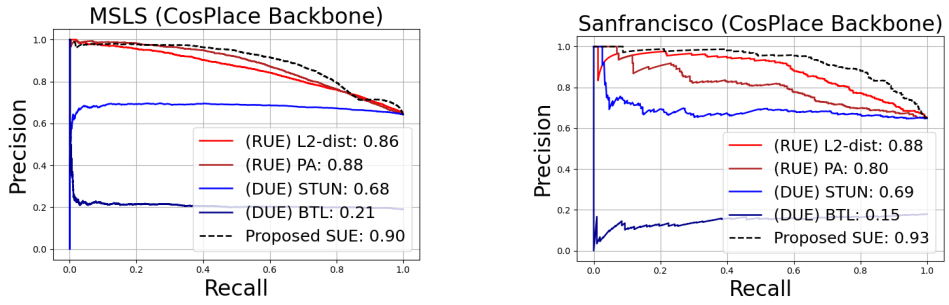
Weigh.	Pitts.	San.	Stlu.	Eyn.	MSLS	Avg
Uniform	0.81	0.77	0.67	0.77	0.49	0.70
SUE	**0.94**	**0.84**	**0.88**	**0.93**	**0.77**	**0.87**

Table 5.4: SUE weighs the contribution of the nearest neighbor poses based on the distance in the feature space with an exponentially decaying function. This performs better than uniform weighing of the variance of the reference poses.



(a) (b)

Figure 5.7: Effect of changing SUE's hyper-parameters K and λ on the AUC-PR. For each curve, the other fixed hyper-parameter is chosen as $K = 10$ or $\lambda = 350$.



(a) (b)

Figure 5.8: SUE remains SOTA by changing the backbone feature extractor to CosPlace [221] with no retuning of SUE's hyper-parameters. CosPlace is also used as the backbone for L2-distance and PA-score, but it was not possible to change the backbone for BTL and STUN.

5.3.5 DISCUSSION

The author now makes several recommendations for estimating the image-matching uncertainty in VPR. First, future works evaluating image-matching uncertainty estimation should include diverse baselines such as SUE, even if they are simple. As the intra-category comparison revealed, even a common L2-distance-based image-matching uncertainty estimation may outperform data-driven techniques. Second, aleatoric uncertainty from training data does not necessarily generalize to the test data, so learning-based approaches should consider that perceptual aliasing is not just a property of the image content, but also the reference map at test time. Referring back to the example of the sky in images being ambiguous for an outdoor map; in an indoor map containing just one open-air patio, such images with sky might instead be considered distinctive for their location. Third, while *GV* gives the best uncertainty estimates at the expense of high computational needs, it is still susceptible to aleatoric uncertainty within the image, as repetitive structures, trees, and walls may also

lead to incorrect matches of local features. In VPR, *GV* methods can still benefit from complementary uncertainty estimates provided by other methods, such as SUE.

The author also notes some potential **limitations of SUE**. SUE may underestimate the uncertainty if K is too small to retrieve aliased references from multiple locations. Selecting K for maps with mixed scene depths can therefore be challenging. Images in areas with low scene depth will already be perceptually distinct at small spatial offsets, whereas at high scene depth even images further apart may suffer from perceptual aliasing. A K that suffices for small scene depths could be too small for areas with high scene depths. This could be mitigated by dynamically incrementing K till $w_{(K)}$ becomes nearly zero. Now consider reference locations A and B which are perceptually aliased, i.e., all their image descriptors are similar. If A has 1000 references and B has one, even with $K \geq 1001$, SUE will always be confident about queries from either A or B as nearly all retrieved matches are spatially close. The high coverage of A over B thus presents an unwanted confidence bias, unless the chance of visiting A over B at test time is also 1000× higher. Nevertheless, the author has shown that despite these assumptions SUE performs well on many real-world datasets.

5

5.3.6 A PROBABILISTIC VIEW OF SUE

The author here presents a probabilistic view of SUE, which will help formulate a modified version in section 5.3.7 to account for different spatial distributions of queries and references.

Consider $M \in \{1, \cdots, N\}$ as a stochastic 'match' variable that indicates which of the N references is a true reference. So, $M = i$ would mean reference i is the 'true' match for the query. Then $\mathsf{p}(M = i)$ expresses the prior belief that any reference i could be the true reference.

Assuming that some reference i is the true reference, $M = i$, then the observed query feature f_q can be expected to be similar to the reference feature f_i, with some homoscedastic Gaussian noise or variation added to all feature dimensions,

$$\mathsf{p}(f_q|M = i) = \mathbf{N}(f_q|f_{(i)}, \Sigma_f) \tag{5.4}$$

$$\propto e^{-\lambda \cdot \|f_q - f_{(i)}\|_2} \tag{5.5}$$

$$\propto w_{(i)}. \tag{5.6}$$

So, the weight term of Equation (5.3) can be considered as the non-normalized likelihood term. Note that the hyperparameter λ subsumes the noise parameter Σ_f.

Through Bayes' rule, we can express the posterior belief over M given the query feature as

$$\mathsf{p}(M|f_q) = \frac{\mathsf{p}(f_q|M)\mathsf{p}(M)}{\mathsf{p}(f_q)} = \frac{\mathsf{p}(f_q|M)\mathsf{p}(M)}{\sum_j \mathsf{p}(f_q|M = j)\mathsf{p}(M = j)}. \tag{5.7}$$

With a uniform prior ($\mathsf{p}(M) = 1/N$) that indicates equal probability for all references, we can see that the posterior reduces to $\mathsf{p}(M|f_q) = w_{(i)}/\sum_j w_{(j)}$, since the constant of the prior factors out in the numerator and denominator.

If we now assume that our VPR technique is reasonable, and that the query position should be located at the 'true' reference, then we can express the expected query position,

given our belief on the match of each reference, i.e.,

$$\mathbb{E}[p_{(M)}|f_q] = \sum_i \left[\mathsf{p}(M=i|f_q)p_{(i)}\right] \tag{5.8}$$

$$= \mu_p \tag{5.9}$$

Here we recognize Equation (5.1), assuming the uniform prior $\mathsf{p}(M)$. While this expected pose is not necessarily considered to be representative of the true query pose (it could be an average location between distant visually-matching areas), it does allow us to compute the expected squared pose distance of the true match to the query,

$$\mathbb{E}\left[\|p_{(M)} - \mu_p\|_2 \middle| f_q\right] \approx \text{trace}(\Sigma_p) = s_q, \tag{5.10}$$

where Σ_p is as defined in Equation (5.2) for the uniform prior $\mathsf{p}(M)$. In other words, in SUE s_q estimates the expected (squared) distance between the match's pose and the query pose, thus the smaller s_q the higher the chance is that a match selected according to our posterior belief is within an acceptable distance to the true query pose.

Finally, reference $i' = \text{argmax}_i\, \mathsf{p}(M=i|f_q)$ with the highest posterior probability of being the correct match is selected, which based on the likelihood term (and with uniform prior) will be $i' = 1$, i.e. the nearest neighbor in the feature space.

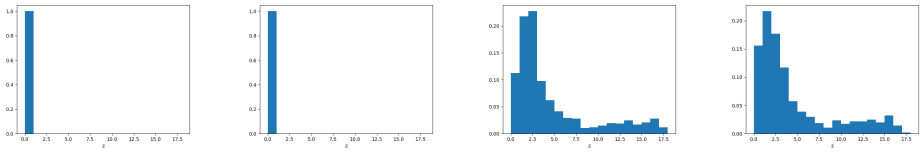
Note that in the above, a uniform prior $\mathsf{p}(M)$ means all references are assumed a-priori equally likely to match the query. In case some areas in the map contain more references than other areas, this also implies a higher prior belief that the query will occur in such a denser sampled area. This 'default' prior is therefore *not* a uniform *spatial* prior over the mapped area, but it assumes that the local spatial density of references in the map is indicative of the probability of a query appearing in such a local region.

5.3.7 SPATIAL DENSITY COMPENSATION FOR DISSIMILAR QUERY/REFERENCE SPATIAL DISTRIBUTIONS

As explained in SUE's potential limitations of section 6.4 and section 5.3.6, the default formulation of SUE assumes that each reference is equally probable to match a query, i.e., a uniform prior $\mathsf{p}(M)$ is assumed. In other words, the query and reference images/poses are expected to be distributed similarly over the map, and the spatial density of the references in an area reflects the assumed prior probability for a query to be located in that area.

To illustrate, consider two perceptually-aliased locations A and B, where location A is represented by 100 images and location B by one image. If a query occurs at A or B, SUE's uncertainty estimate as currently formulated in Equation (5.2) will be low, since the many references at location A will all agree on low spatial variance, while the contribution of distant references at location B are 100× less. This high confidence could be desired if location A is also 100× more likely to be visited at query-time than location B (i.e. the uniform $\mathsf{p}(M)$ holds, so the spatial density of the references reflects a spatial prior of a query's location). However, this prior could also be undesired if we expect queries at A and B are equally likely to occur, irrespective of the reference density. Ultimately, what is desired depends on the application and data collection procedure.

In case the uniform prior $\mathsf{p}(M)$ over references is undesired, we can substitute it with a different prior in the equations of section 5.3.6. Specifically, in Equation (5.7) the likelihood



(a) Pittsburgh query (b) Pittsburgh reference (c) Stlucia query (d) Stlucia reference

Figure 5.9: The density of queries and references is depicted using the distance (z) of each query/ref to its nearest neighbour ($k = 1$) in the pose space. Queries and references in Pittsburgh dataset are highly dense and hence uniformly spatially distributed. The queries and references are non-uniformly (albeit similarly) spatially distributed in the sparser Stlucia dataset.

terms should *not* be multiplied with a constant prior term (which cancelled out in the numerator and denominator). Still, it may be more convenient to express the prior over references in terms of a *spatial prior for the query*. In other words, a reference would be more probable to match if it is in a area where the query is more probable to occur, while a reference would be less probable if there are more other references in the same spatial area. Let $\mathsf{p}_q(p)$ denote the desired spatial prior for the query to be at a pose p, and $\mathsf{p}_r(p)$ denote the spatial density of the references at a pose p, then

$$\mathsf{p}(M = i) \propto \frac{\mathsf{p}_q(p_{(i)})}{\mathsf{p}_r(p_{(i)})}. \tag{5.11}$$

This will be referred to as *spatial density compensation*. In practice, we can thus compensate SUE for a desired spatial prior by multiplying the reference weight $w_{(i)}$ with a term (proportional to) the desired prior $\mathsf{p}(M)$. Note from Equation (5.11) that if the spatial distributions of queries and references are assumed equal, we again obtain that $\mathsf{p}(M)$ is uniform, as is the case for the default SUE formulation.

5.3.8 VALIDATING SPATIAL DENSITY COMPENSATION

In this section, the spatial density compensation concept of adjusting SUE will be tested, as explained in section 5.3.7.

Applying a uniform spatial prior for the query Let's assume the spatial density of query poses is uniform, so all query poses within the map are equally likely, in which case term $\mathsf{p}_q(p)$ becomes a constant (and thus will cancel out when normalizing the weights).

The spatial density of the references $\mathsf{p}_r(p)$ can be estimated from the finite samples of poses in the reference set. We can for instance model the spatial density of references by simply taking the distance $z_{(i)}$ of the reference i to its k-th nearest neighbor in the *pose space*, such that the area $z_{(i)}^2$ is inversely proportional to the local density of the reference i, i.e., $\mathsf{p}_r(p_{(i)}) \propto 1/z_{(i)}^2$. Hyperparameter k regularizes the smoothness of the estimated reference pose density.

We can now see that $\mathsf{p}(M = i) \propto z_{(i)}^2$, thus the density compensated SUE for this uniform

Compensation	Pitts.	San.	Stlu.	Eyns.	MSLS
none	0.94	0.84	0.89	0.93	0.76
$k = 1$	0.94	0.84	0.82	0.93	0.76
$k = 3$	0.94	0.84	0.84	0.93	0.77
$k = 10$	0.94	0.81	0.85	0.92	0.77

Table 5.5: SUE's AUC-PR with reference density compensation.

spatial prior for query poses is obtained by re-weighing Equation (5.3) with $z^2_{(i)}$, i.e.,

$$w_{(i)} = e^{-\lambda \cdot d_{(i)}} \cdot z^2_{(i)}. \tag{5.12}$$

Do common datasets have a uniform query distribution? The above formulation of spatial density is used to study the properties in the existing VPR datasets. First, the author found that most of the datasets *do* have a mostly uniform spatial distribution for both queries and references, except the Stlucia dataset. Fig. 5.9 illustrates the distribution of distances to the $k = 1$ nearest neighbors for the Pittsburgh and Stlucia datasets. Second, it can be concluded that the assumption that references and queries have a similar spatial distribution *does hold* in common VPR dataset, hence SUE's default formulation with uniform reference prior is reasonable.

To properly validate the density compensation concept of section 5.3.7, a modified version of the Stlucia data is also created such that queries and reference actually do have a *different* spatial distribution. The Stlucia queries are greedily subsampled such that the spatial density of the resampled queries is uniform.

Does assuming a uniform query distribution help? Finally, the author will test the density compensated SUE of Equation (5.12) on the VPR datasets for different choices of k, see Table 5.5.

Since queries and references of datasets other than Stlucia are already uniformly distributed spatially, the table confirms that density compensation does not lead to any major effect on SUE's performance. It can also be seen that for the (unmodified) Stlucia dataset, density compensation actually *hurts* performance because the queries and references are in fact *non*-uniformly and similarly distributed. The default uniform prior assumption of SUE is therefore better suited for Stlucia.

However, if the density-compensated SUE is tested on the modified Stlucia dataset where queries are in fact uniformly spatially distributed while the references are not, then there is no benefit observed over the default SUE as shown in Table 5.6. In this case, the spatial prior of density compensated SUE does hold, whereas the default SUE assumption that queries and references are similarly distributed does not.

In conclusion, whether spatial density compensation is needed depends on the specifc spatial distributions of the references and queries in a dataset. For the studied VPR benchmark datasets that represent densely collected queries and references, the default assumption of SUE that their spatial distributions are similar hold. Still, in applications where we can expect that queries and references are distributed differently, then additional density compensation can be helpful. The formulation of spatial density compensation can be motivated from

z	none	k=1	k=3	k=5	k=8	k=10
$8-9$	0.92	**0.96**	0.96	0.96	0.94	0.94
$10-11$	0.68	**0.76**	0.73	0.7	0.71	0.69

Table 5.6: SUE's AUC-PR with reference density compensation using different values of k on the Stlucia dataset when the queries are resampled to have a close to uniform spatial density (e.g., $z = 8-9$). Reference density compensation helps SUE when queries are spatially uniformly distributed and references are non-uniformly distributed. Best across the columns is in Bold.

a probabilistic view on SUE. Future work can investigate better estimates for query and reference density for non-uniformly distributed data to further improve SUE.

5.3.9 SUPPLEMENTARY RESULTS

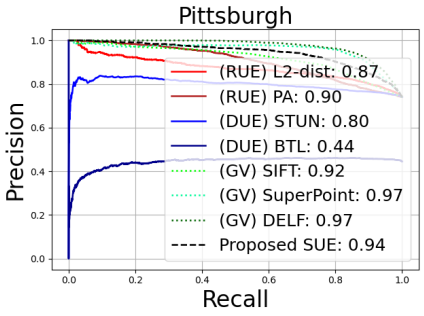
Precision-Recall curves In addition to the Precision-Recall curves of the Pittsburgh dataset in Fig. 5.1, the PR-curves for the remainder five datasets are shown in Fig. 5.10. SUE outperforms the methods in the *RUE* and *DUE* categories on all datasets. *GV* remains the overall state-of-the-art, albeit at a two to three times higher computational cost.

Complementing geometric verification Fig. 5.11 further shows the generalization of the SVM trained on the Pittsburgh dataset to other datasets. For all these datasets, the relation of the SUE uncertainty with DELF-RANSAC leads to complementarity with queries in the bottom-left of the plot that can be linearly separated.

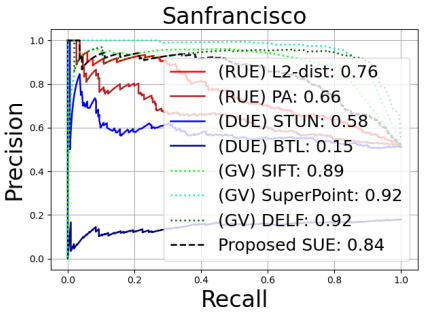
SUE combined with other uncertainty estimates For completeness, the author shows the combination of other uncertainty estimation methods with SUE in Fig. 5.12. Most of the queries that can be classified as true- or false-positives by other methods can already be classified using only SUE. The author hypothesizes that this is because of SUE's similarity to BTL and STUN which also estimate the aleatoric uncertainty, and since SUE already uses the L2-distance and nearest neighbours in its uncertainty estimate.

5.3.10 QUALITATIVE RESULTS

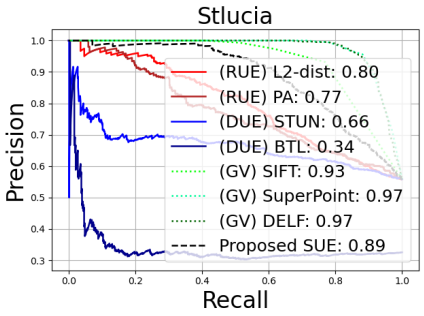
More examples of queries are shown with their corresponding nearest neighbors ranked with the uncertainties computed by the different types of uncertainty estimation methods in Fig. 5.13. The set of randomly chosen queries is kept the same for all the methods. These examples further indicate what each method is sensitive to for uncertainty estimation.



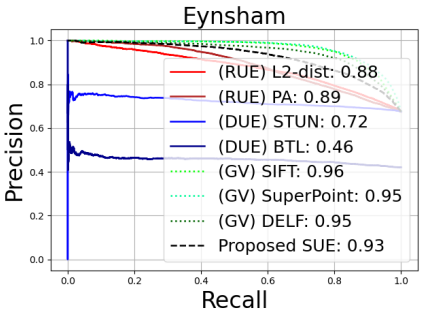
(a)



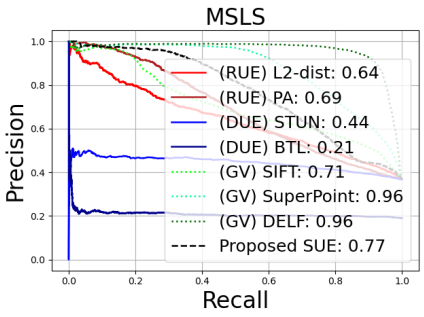
(b)



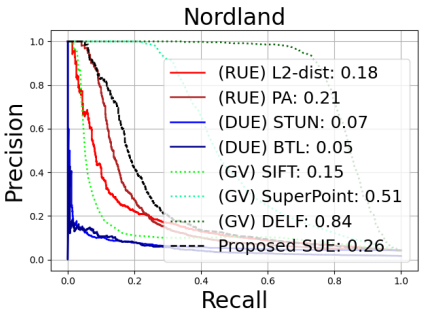
(c)



(d)



(e)



(f)

Figure 5.10: The precision-recall curves on the six datasets using SUE and other baselines. SUE outperforms the existing methods within the efficient category on all datasets. Note how an L2-based retrieval uncertainty outperforms the data-driven aleatoric uncertainty estimated in BTL and STUN.

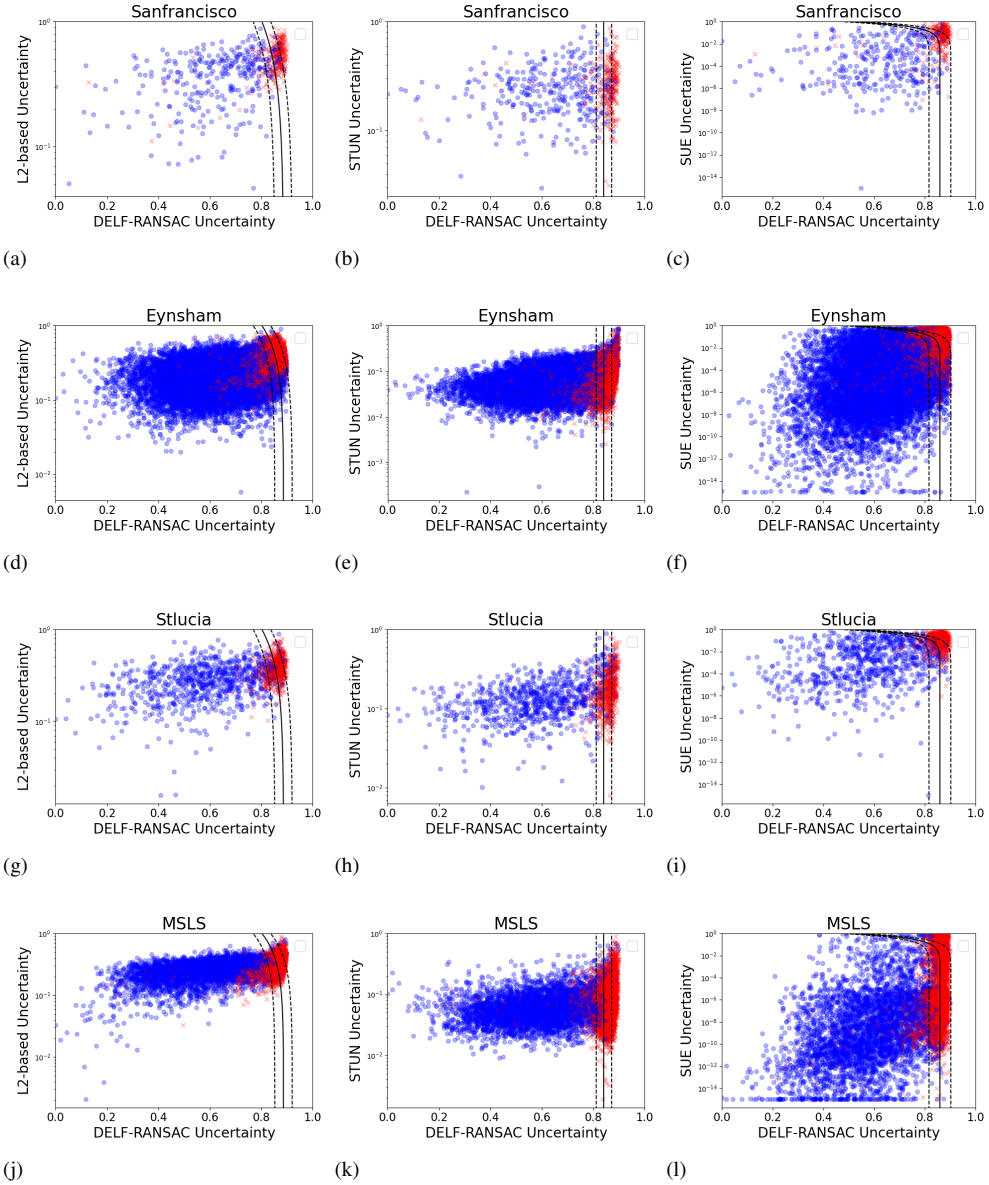


Figure 5.11: The relation of L2-based uncertainty, STUN, and SUE with geometric verification uncertainty. The SVM boundaries are learned on the Pittsburgh dataset only. Each point represents a query, and the color indicates whether it is a true-positive (Blue) or a false-positive (Red). The linear SVM boundaries are shown as black lines, while the dashed lines are the SVM margins. The combination of SUE with geometric-verification leads to more correctly matched queries in the bottom right (where SUE is certain but *GV* is uncertain) of the plots identifying complementarity. For better visualization, the vertical scale is in log-space, due to which the SVM boundaries appear non-linear to the reader but are linear.

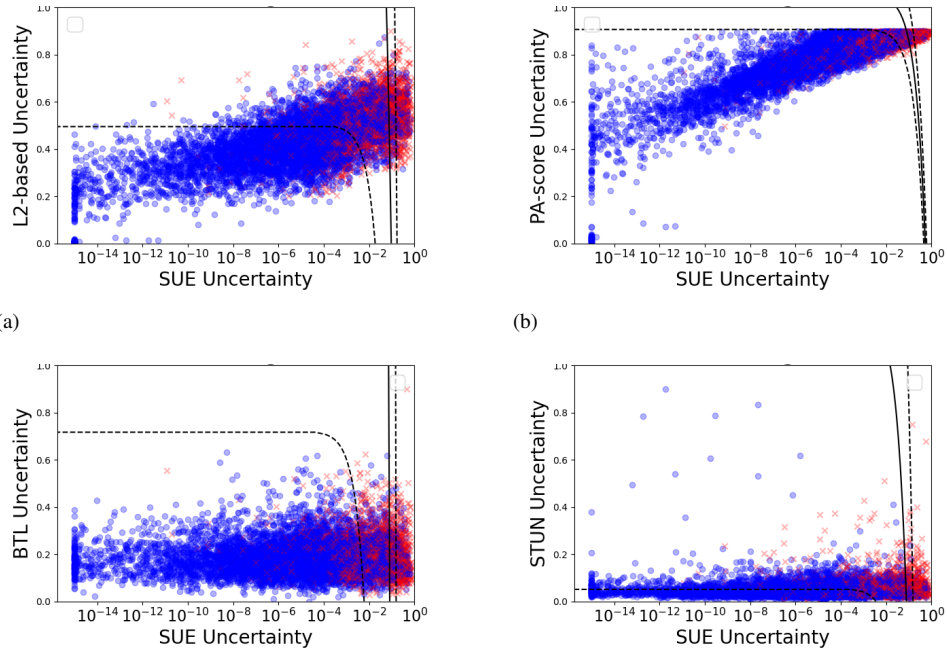


Figure 5.12: The relation of L2-based, PA-score, BTL, and STUN uncertainties with SUE uncertainty. Each point represents a query, and the color indicates whether it is a true-positive (Blue) or a false-positive (Red). The linear SVM boundaries are shown as black lines, while the dashed lines are the SVM margins. As indicated by the near-vertical decision boundaries, most of the queries that can be classified as true- or false-positives by other methods can also be classified by SUE, and we do not see much complementarity.

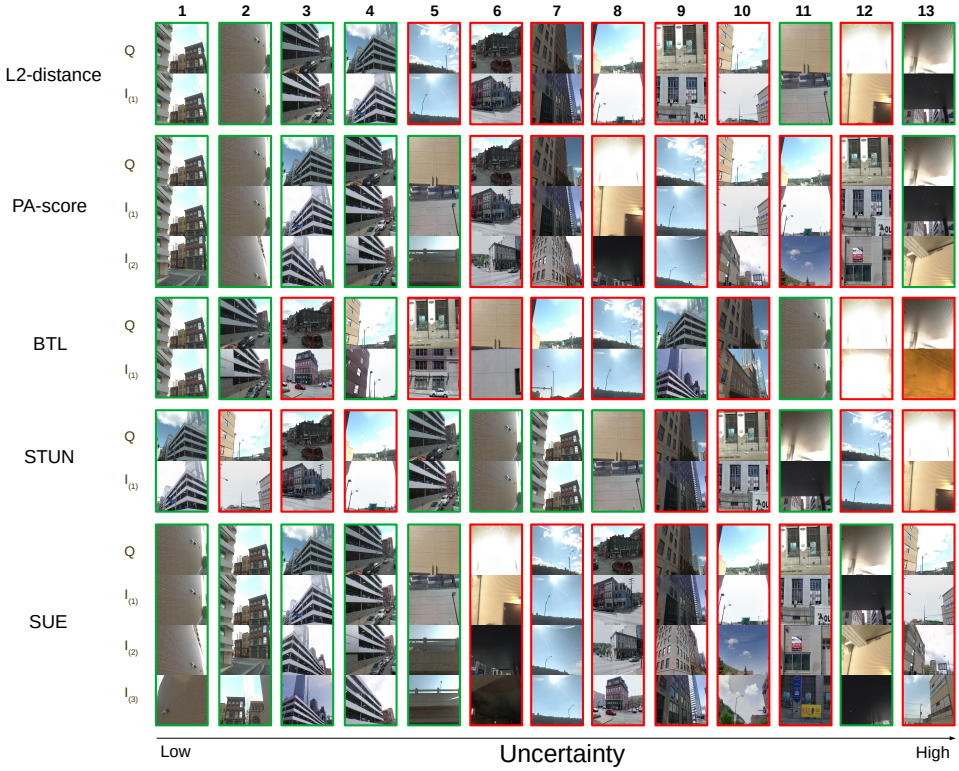


Figure 5.13: Exemplar matched/mismatched queries are ranked with different types of estimated uncertainties in the Pittsburgh dataset. Note that the set of chosen queries is the same for all types of uncertainty estimation methods. $I_{(n)}$ denotes the nearest neighbor where the subscript n denotes its rank. The number of nearest neighbors shown relates to the corresponding number needed by each method (e.g. PA-score requires two nearest neighbors). The retrieved nearest neighbors for BTL are different than other methods due to the different feature encoder. A good uncertainty estimation method when used for ordering would rank correct matches to the left and incorrect matches to the right of the reader. The query image in column 12 of SUE depicts the failure case of SUE, where the perceptually aliased nearest neighbors are geographically far-apart leading to high uncertainty but the best match is still the correct match.

5.4 CONCLUSIONS OF THE CHAPTER

The author has compared different approaches for estimating the image-matching uncertainty in VPR, which provided (surprising) insights into this task, e.g., existing methods that learn aleatoric uncertainty from the training dataset often do not generalize well to the reference map at test time, and the common L2-distance in the feature space can be a more reliable indicator of matching uncertainty. The author has shown that matching uncertainty in VPR is tightly related to the reference set at test time. The proposed new baseline SUE uniquely considers the spatial locations of the references, and outperforms all but the computationally expensive geometric verification. Its uncertainty estimates complement those of geometric verification. The choices for SUE's hyper-parameters generalize for most queries across the tested datasets.

We have by now discussed the challenging tasks of domain-generalization and uncertainty estimation in VPR, and looked at potential solutions proposed by the author. Nevertheless, even if a VPR method works universally across domains and provides perfect uncertainty estimates, it still may not be enough for accurate VPR-based localization. The next chapter will expand on this topic, and we will see potential solutions to this, once again, by exploiting the test-time reference map in VPR.

6

Continuous Place Descriptor Regression (CoPR) in the Test-time Reference Map

This chapter is based on *M. Zaffar. CoPR: Towards Accurate Visual Localization With Continuous Place-descriptor Regression, T-RO, 2023.* [38].

Author contributions: Mubariz Zaffar proposed and implemented the method, performed the experiments, and took the lead in writing and presenting. Julian Kooij provided suggestions on the methodological formulation, experimental design and technical writing. Liangliang Nan provided feedback on the writing and visualizations.

6.1 OVERVIEW

The previous chapters discussed how domain generalization and uncertainty estimation are two important tasks for VPR, and how the test-time reference map can be exploited to tackle these challenges. However, even if a VPR method is universal and provides perfect uncertainty estimation, it is unclear whether it can lead to accurate VPR-based localization. Accurate Visual-based Localization (VBL), i.e., to localize a robot in a map using only an image as the input from a robot's camera [239] is a key requirement in robotics and computer vision. In this section, we are going to briefly and broadly recap the different types of VBL and the role of VPR within it.

Various parallel research directions have emerged within VBL. A top-level distinction can be made between purely image-based approaches and 3D structure-based approaches. The former are simple and efficient but have lower localization accuracy, while the latter are more accurate at the cost of increased computation complexity and maintenance effort [4]. Purely image-based approaches could be further divided into Visual Place Recognition (VPR) [80], Absolute Pose Regression (APR) [19], and Relative Pose Estimation (RPE) [21]. Given their efficiency and scalability, VPR techniques are often used in robotics for loop closure detection or 3D reconstruction. However, improving their performance remains an ongoing research challenge [5] [8].

VPR, as we know by now, is the task of finding for a query image the best matching reference image from a set of pre-recorded geo-tagged reference images (i.e., the reference map) [233]. Each reference image is considered a 'place', and the geo-location of the best-matched reference is then the estimated location ('place') of the query image. Whereas VPR relies on image-retrieval, in APR a neural network directly regresses the global coordinates for a query image, and the map is implicitly represented by the network weights. However, such APR methods do not generalize across viewpoints, as has been studied by Sattler et al. [29]. RPE, on the other hand, operates on two images with assumed nearby viewpoints, and estimates from the overlapping image contents the relative translation and orientation between their corresponding camera coordinate frames. Since VPR performs coarse global localization, and RPE performs fine-grained localization by assuming coarse localization is solved, both techniques are often combined in the multi-stage approach, referred to as Coarse-to-Fine localization (CtF) [21] [240] [154]. RPE is therefore not an alternative to VPR, but a refinement step that is only successful if VPR was able to retrieve a nearby reference.

VPR remains less accurate than structure-based and CtF approaches [73], with a crucial reason being the discrete nature of the reference map in VPR. When a query image appears between two anchor locations in the reference map, a VPR system could at best only match this to the nearest spatial anchor location, incurring some minimal Euclidean distance error. This can become worse when query images and existing reference images span the same area but at offsets of parallel lines, as shown in Fig. 6.1. Therefore, the author seeks to add more references to the map (such as the blue poses in Fig. 6.1), a notion referred to as *map densification*. A trivial but often impractical solution to densification is to collect more reference images. Alternatively, densification could be achieved by creating a 3D model of the environment and rendering images at novel poses. However, creating and maintaining up-to-date 3D models is computationally and storage-wise expensive, and the resulting images are not photo-realistic [29, 241].

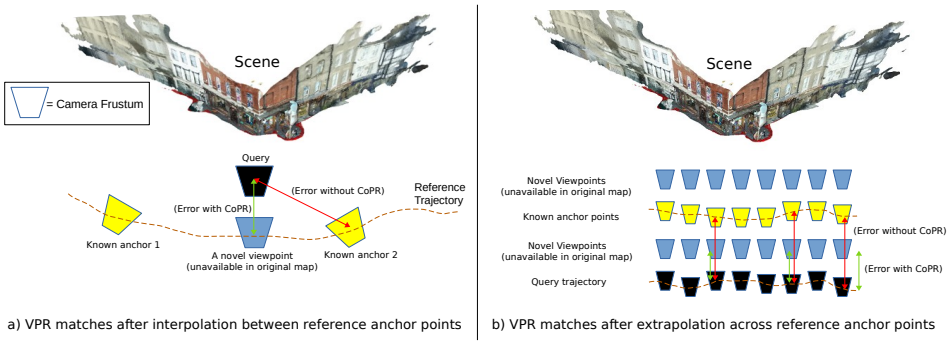


Figure 6.1: The discrete treatment of VPR that leads to lower localization accuracy. Provided that only the *yellow* anchor reference poses are available in the map, the *black* query images could only be matched as close as possible to the base error. Regressing descriptors for the *blue* target viewpoints using interpolation or extrapolation given anchor reference descriptors could lead to improved localization accuracy for query images in VPR and thus reduce the base error. The scene shown in this figure is taken from the work of Sattler et al. [29].

Since the VPR reference maps comprise compact feature descriptors of images, the author suggests performing map densification in the *feature space* rather than the image space. Continuous Place-descriptor Regression (CoPR) is proposed in feature space for VPR map densification.[1] Since in CtF the RPE step assumes the initial VPR step was performed correctly, we note that improving VPR could also address CtF errors that cannot be corrected by RPE, as we will also see in this chapter.

The author argues for two requirements to benefit from such map densification: 1) a method of regressing meaningful feature descriptors for VPR at novel target viewpoints given anchor point feature descriptors, and 2) an image-retrieval system that is viewpoint-variant and therefore could utilize the regressed descriptors at target viewpoints. Furthermore, the model for descriptor regression should only need existing anchor descriptors and relative poses between anchor locations and target viewpoints, at its input, and it should not require images of the scene from target viewpoints or expensive scene reconstruction [29].

To study the problem of descriptor regression, the author further considers two possible schemes: interpolation and extrapolation. Both of these are relevant for map densification, where *interpolation* (Fig. 6.1a) refers to interpolating to an intermediate location between some anchor points on the reference trajectory, while *extrapolation* (Fig. 6.1b) refers to regressing descriptors around a given anchor reference pose. Since interpolation could even be performed using averaging of the nearest anchor points along the trajectory, i.e., by simply following the trend in the local feature space, it is expected to be an easier problem to solve than extrapolation. Extrapolation, on the other hand, is a more important requirement for map densification, because it enables us to potentially regress descriptors at or close to the query. Interpolation can at best only densify within an existing reference trajectory.

Finally, for a VPR system to benefit from map densification, it needs to retrieve the

[1]This discrete nature of the reference map is also problematic for APR as reported by [29]. The author hypothesizes that APR could also benefit from map densification via descriptor regression, but this aspect is not explored in this chapter, and the scope is limited only to VPR.

Euclidean closest match in the physical space as the best match in the feature space. This is not enforced in VPR techniques trained with triplet-loss [80], classification-loss [78], and ranking-based-loss [96], where the correct/incorrect ground-truth match is discrete (leading to viewpoint invariance), instead of the continuous ground-truth in distance-based loss [160]. If a VPR technique is viewpoint-invariant, both the blue trajectories in Fig. 6.1b would be incorrectly considered equally valid. Thus, it is hypothesized that map densification and viewpoint variance should work hand in hand to make VPR-based localization more accurate. It is shown that a highly viewpoint-variant VPR technique in a densified reference map leads to the highest localization accuracy, amongst all the combinations originating from the different feature encoders and levels of map densification.

In summary, the contributions of this chapter are as follows:

1. The author investigates Continuous Place-descriptor Regression (CoPR) to densify a sparse VPR map through either interpolation or extrapolation of the feature descriptors to target poses, without requiring any new measurements (i.e., reference images).

2. Linear regression-based techniques and a non-linear deep neural network are proposed for map densification, and the author demonstrates improvement in localization accuracy on three existing public datasets.

3. It is reported that different feature encoders can benefit from map densification, and the best performance is achieved by using the most viewpoint-variant descriptors in a densified map.

4. The author discusses the VPR failure cases where RPE cannot recover the correct pose without CoPR, highlighting the complementarity of these approaches for improving VBL accuracy. The existence of such cases is demonstrated with real-world data.

6.2 METHODOLOGY

The formal definition of VPR is first recapped here within the context of this chapter. Given a set of reference images with known poses, VPR constructs a map $M = (R, P)$, where R is a set of reference descriptors, such that $f_i \in R$ is an N-dimensional feature descriptor with a corresponding pose $p_i \in P$. Each feature descriptor $f_i = G(I_i)$ is obtained from a reference image I_i using an already trained and fixed feature extractor G, typically a neural network. The pose p_i is a 6 degree-of-freedom pose that specifies the location as a translation vector $t_i = (x, y, z)$, and a quaternion vector o_i specifying the 3D orientation.

At test time, the objective is to find the pose p_q of a query image I_q, for which the query descriptor $f_q = G(I_q)$ is computed. The descriptor f_q is matched to all the reference descriptors in the set R, and the Nearest Neighbor (NN) match $r_{nn} = \text{argmin}_{r \in R} \|f_r - f_q\|_2$ is retrieved. The pose of the query image is then considered the same as that of the retrieved reference descriptor, i.e., $p_q = p_{nn}$. Ideally, the feature descriptors are constructed such that the resulting Euclidean translation error $e = \|t_q - t_{nn}\|_2$ is minimal. Hence, the assumption $p_q = p_{nn}$ is essentially an approximation $p_q \approx p_{nn}$, and would only be true in the unlikely event that the query is collected at the same pose as that of the retrieved reference in the map. Thus, the expected error $\mathbb{E}[e]$ is a non-zero *base error* of a VPR system. This base error

is directly affected by the sparseness in the reference map: the further apart the reference samples are, the higher the base error could be[2]. Therefore, this chapter proposes to apply map densification for VPR as shown in Fig. 6.1.

6.2.1 MAP DENSIFICATION

To reduce the base error, the author seeks to extend the number of descriptors and poses in a given sparse map M_{sparse}. Since collecting more reference images is not always possible, the aim is to perform densification using only existing reference descriptors in M_{sparse} without the need to collect more images at novel viewpoints. Such densification in feature space also has computational benefits since image-description is more computationally expensive than descriptor-regression, as shown later in sub-section 6.3.8. Concretely, the author proposes to densify a sparse map $M_{sparse} = (R, P)$ by defining a set of target poses P' for which the corresponding descriptors R' are predicted via Continuous Place Descriptor Regression (CoPR) using one or more existing reference descriptors in R which we will refer to as *anchor descriptors*. The resulting densified map $M_{dense} = (R \cup R', P \cup P')$ thus extends the original map M_{sparse} with the newly regressed target references.

Different strategies could be employed to define (a) which set of target poses P' to regress to, and (b) how to regress the descriptors for a target pose using the available anchor descriptors. The author here explores two specific strategies for defining the set P', namely (1) interpolating between the anchor points on the reference trajectory. and (2) extrapolating to nearby poses of an anchor pose that do not necessarily lie along the reference trajectory. Regression approaches will be discussed later in sub-section 6.2.2.

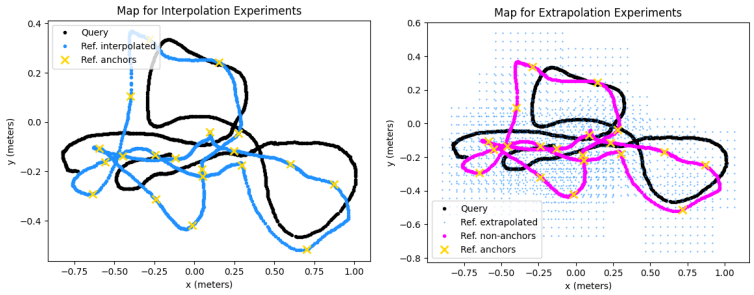


Figure 6.2: The test setup for the interpolation and extrapolation experiments on the Heads scene of the 7-scenes dataset in 2D. The anchor reference points are to be used by regression techniques to interpolate/extrapolate descriptors at target poses. Since, in the case of extrapolation, sub-sampling is not done along the reference trajectory as in interpolation, there are non-anchor reference points in the extrapolation experiment but not in the interpolation experiment.

The **interpolation scheme** assumes that the references in the sparse map are obtained in a sequence. Additional poses P' can be selected along the trajectory in between the poses available in P. Hence, any two subsequent references $a1 \in R$ and $a2 \in R$ can be selected as

[2]Clearly, if the query images appear at the exact same spot as that of the reference trajectory, map densification would not help. This, however, is highly unlikely and unrealistic in real-world situations as evident in existing VPR datasets. [8]

anchors, and one or more new target poses p_{new} can be selected on the path between the anchor poses p_{a1} and p_{a2}.

In the **extrapolation scheme**, the set of target extrapolation poses P' is selected in the vicinity of the poses in P, but not necessarily on a path between them. One possibility is to generate these target poses in a uniform grid within a certain distance threshold around each anchor. Another possibility is to define a single global uniform grid, and only evaluate grid points using the nearest anchor points (within some distance threshold) similar to [29]. The former approach leads to a denser grid, although it is globally non-uniform.

Examples of the reference, query, and target poses are shown in Fig. 6.2 to illustrate interpolation and extrapolation for map densification on the 7-scenes dataset [242].

6.2.2 DESCRIPTOR REGRESSION STRATEGIES

Several strategies are considered to predict a new descriptor $f_{new} \in R'$ for a given target pose $p_{new} \in P'$ and the sparse reference map M_{sparse}, which could be applied to the extrapolation and/or interpolation tasks. In principle, a regression method fits a model to express the dependent variable(s) as a function of the independent variables, thereby capturing the local trend in the space around the fitted samples. For feature descriptor regression, the objective is to express the feature space as a function of the pose. Since this feature space is latent, it is unclear to what extent we can assume it to be globally or locally linear for changing pose; hence, the author considers both linear and non-linear regression techniques for CoPR, as follows.

LINEAR INTERPOLATION

The simplest strategy only applies to interpolation, where only the translation is used and not the orientation of each pose. The aim is to predict the descriptor for an intermediate translation between two known translations. The target descriptor in this case is a linear weighted combination of its two anchors,

$$f_{new} = (1-\alpha_{a1}) \times f_{a1} + (1-\alpha_{a2}) \times f_{a2}, \tag{6.1}$$

$$\alpha_{a1} = \beta_1 \;/\; (\beta_1 + \beta_2), \tag{6.2}$$

$$\alpha_{a2} = \beta_2 \;/\; (\beta_1 + \beta_2), \tag{6.3}$$

where $\beta_1 = \|t_{new} - t_{a1}\|_2$, $\beta_2 = \|t_{new} - t_{a2}\|_2$, and f_{a1}, f_{a2} are the two anchor feature descriptors.

LINEAR REGRESSION USING LOCAL PLANE FIT

As a second approach, the author investigates a local plane fit to consider more anchors and allow extrapolation too. This also only uses the translation and not the complete pose. Given the target translation t_{new}, the O Nearest Neighbor anchor points from M_{sparse} in terms of Euclidean translation distance are selected. For each descriptor dimension, a linear plane is least-squares fitted on the anchor values, and the plane is evaluated at the translations of the target t_{new} to regress f_{new}. This linear regression is abstractly depicted in Fig. 6.3 for a single feature dimension (f) in a two-dimensional pose space (x and y). Note that a more complex polynomial or spline regression could be used too, but the author limits his approach to linear regression here as the most canonical implementation of this general approach.

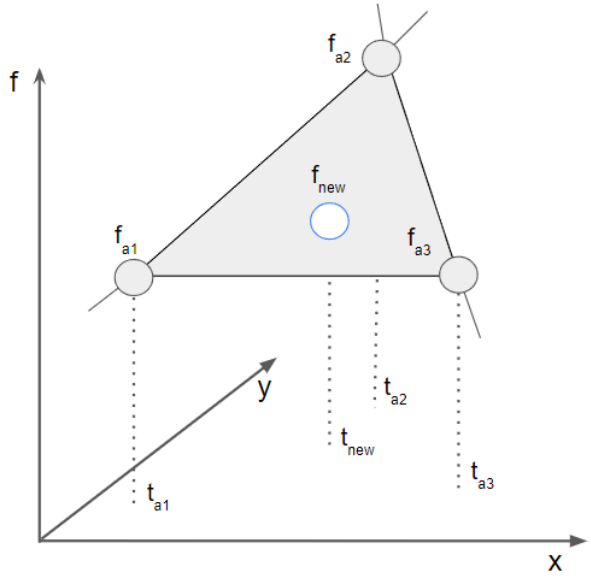


Figure 6.3: A locally-fit plane given three anchor points in a two-dimensional world. Note that this plane is for a single feature dimension, so in practice, there will be N such planes.

NON-LINEAR REGRESSION NETWORK

In this strategy, the author directly regresses $f_{new} = H(f_a, \Delta p)$ from a single anchor descriptor f_a, and the relative pose Δp specifying the translation difference and the quaternion rotation between the anchor pose p_a and the target pose p_{new}. As non-linear descriptor regressor H, a fully-connected deep neural network is used consisting of 7 hidden layers with a GeLU [243] activation. The input to the network is the N dimensional anchor feature descriptor f_a and the relative pose Δp stacked together, while the output is the N dimensional target feature descriptor f_{new} at the pose p_{new}. The dimensionality of the input layer and hidden layers is the same, i.e., $N+7$, as the relative pose vector Δp has a length of 7, while the output layer has only N dimensions. This network is shown in Fig. 6.4. In preliminary experiments on Microsoft 7-scenes (see sub-section 6.3.1), the author explored other activations and using fewer or more layers. GeLU was found to work the best, and the network would overfit with more than 7 layers.

Given a pre-trained and fixed encoder G for computing feature descriptors, the non-linear regression network is trained on available descriptor pairs (e.g., an anchor descriptor f_a and a ground-truth target descriptor f_{gt}) with known relative pose Δp between them, and a mean-squared error loss,

$$L_{MSE} = \|H(f_a, \Delta p) - f_{gt}\|_2. \tag{6.4}$$

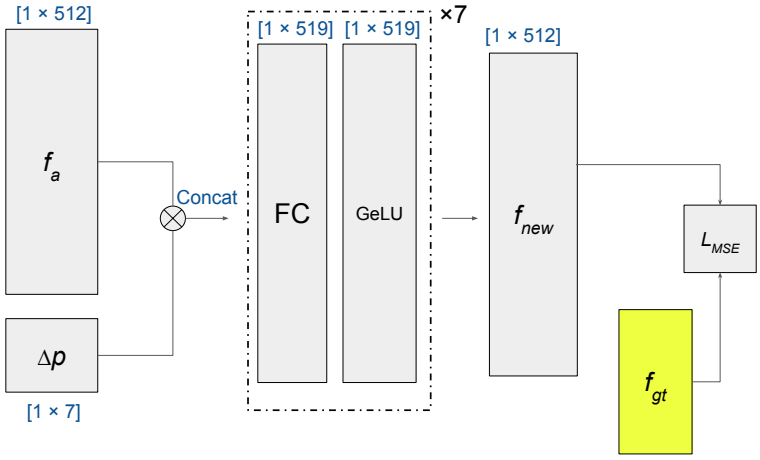


Figure 6.4: The non-linear deep learning based model H that is trained to regress the descriptor f_{new} at a target location. The input is an anchor reference descriptor f_a and the relative pose Δp between the anchor location p_a and the target location p_{new}.

6.2.3 LOSSES FOR THE FEATURE ENCODER

Next, the choice for the training loss of the feature encoder G is discussed, since the feature space is key for the general localization quality, and also defines the complexity of the regression task that map densification should solve. The feature encoder G takes as input an image I and computes its N dimensional feature descriptor f_I. The author will compare three different training strategies, namely training with a triplet loss [80], an RPE loss [21], and a distance-based loss [160], which are shortly summarized here.

For training with a **triplet loss**, the network computes N dimensional feature descriptors $\{f_q, f_p, f_n\}$ for three images $\{I_q, I_p, I_n\}$: a query I_q, a positive match I_p with varied viewpoint and a negative match I_n that represent a different scene/place. Each of these three N dimensional feature descriptors is then normalized and penalized with a triplet loss. The triplet loss is the same as that of [80] which penalizes the network given a Euclidean distance function $d_f(f_1, f_2) = \|f_1 - f_2\|_2$ and a margin m with a triplet loss,

$$L_{triplet} = max\{d_f(f_q, f_p) - d_f(f_q, f_n) + m, 0\}. \tag{6.5}$$

For the **RPE loss** [21], f_q and f_p are stacked together and passed through a relative-pose regressor consisting of fully-connected layers to output the estimated 6-DoF relative pose Δp_{est} between the two input images. The network is trained with a mean-squared error loss, i.e.

$$L_{relative} = \|\Delta p_{est} - \Delta p_{gt}\|_2, \tag{6.6}$$

given the ground-truth relative pose Δp_{gt}. This is the same network as that of Laskar et al. [21]. To regress the relative pose Δp_{est} correctly, the network has to encode viewpoint information in the feature descriptors $\{f_q, f_p\}$. Nevertheless, this relative pose-based loss does not explicitly force the network to encode representations that encourage the closest descriptor in 3D physical space to be the closest in feature space.

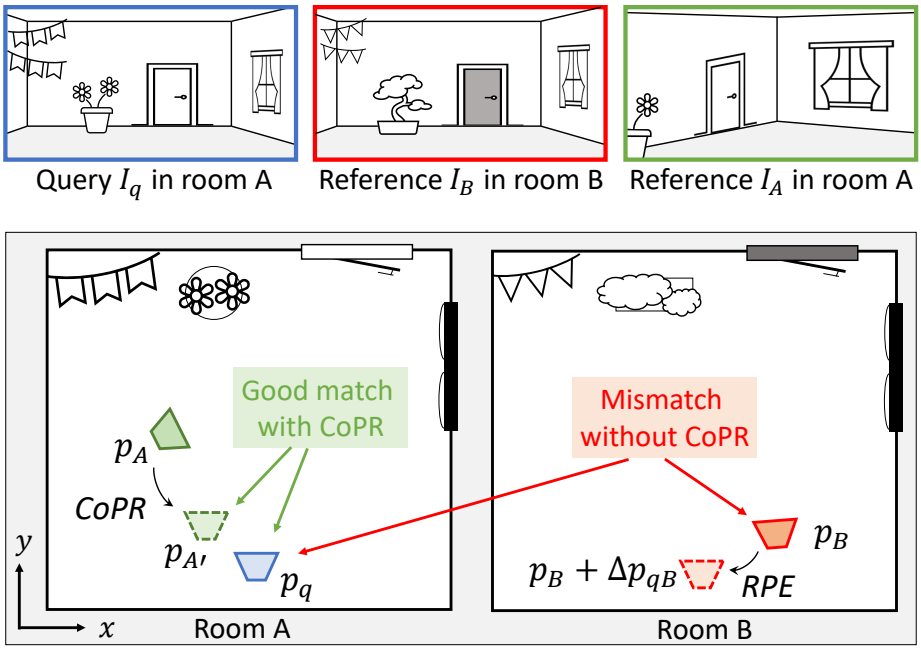


Figure 6.5: Perceptual aliasing of rooms A and B: query I_q in room A appears more similar to reference I_B in room B than to reference I_A in correct room A. If VPR retrieves the wrong reference f_B for f_q, RPE between f_B and f_q cannot correct this: the 'apparent' difference between the query pose p_q and reference pose p_B is nearly zero. CoPR therefore aims to improve VPR instead by adding references for more diverse poses to the map, e.g. $f_{A'}$ for $p_{A'}$.

Therefore, the third loss is the **distance-based loss** $L_{distance}$ as introduced in the work of Thoma et al. [160],

$$L_{distance} = \|\Delta f - \Delta t\|_2. \tag{6.7}$$

This loss explicitly penalizes the network based on the Euclidean distance Δf between feature descriptors $\{f_q, f_p\}$ and the Euclidean distance between their corresponding ground-truth translation poses Δt.

6.2.4 RELATING COPR TO RELATIVE POSE ESTIMATION

My main focus is the task of VPR for VBL. Nevertheless, map densification can also improve the accuracy of Coarse-to-Fine localization, i.e., VPR plus RPE [21]. This sub-section expands on the methodological relation between CoPR and RPE.

Formally, given two feature descriptors f_1 and f_2 and the relative pose between their corresponding locations Δp, a CoPR strategy as in sub-section 6.2.2 models a function $f_2 = H(f_1, \Delta p)$. In contrast, RPE aims to learn a function $\Delta p = L(f_1, f_2)$. While these two functions H and L appear similar, these approaches have different benefits. A useful property of CoPR is that it can be done offline, thus localization reduces to a single-stage image-retrieval problem at runtime, while RPE is performed online and thus leads to a multi-stage CtF formulation.

A more crucial difference is that RPE assumes its two input images represent the same scene, and thus must rely on the accuracy of the preceding image-retrieval step. Consider a query I_q taken in a scene A, e.g., a room in an office, and a sparse reference map containing various visually similar scenes, e.g., other rooms in the same office (see Fig. 6.5). The image-retrieval system might fail and retrieve a reference f_B from an arbitrarily distant scene ('room') B instead of any nearby reference f_A from the actual scene A, i.e. when $\|f_B - f_q\|_2 < \|f_A - f_q\|_2$. This inability to distinguish such similar scenes is referred to as *perceptual aliasing* [244]. These scenes should ideally all be represented as nearby references in the feature space, but in a sparse reference map some scenes could be underrepresented, and retrieving the best (or even top-k) matches for a query might never include the correct scene. RPE cannot correct such retrieval failures. For instance, a pose difference between correct reference I_A and query I_q (both at room A) could limit the visual overlap between their images, making their descriptors f_A and f_q dissimilar. If the visual content of I_b and I_q appear more similar, their pose difference would *appear* relatively small, even though these are at completely different scenes. Since $\Delta p_{qB} = L(f_q, f_B)$ will just estimate the small *apparent* pose offset, RPE results in an incorrect final pose estimate for the query, $p_B + \Delta p_{qB}$.

By densifying the reference map, the references in room A can be extended to represent more diverse poses. A regressed descriptor $f_{A'}$ at a new pose $p_{A'}$ closer to the query than the original reference p_A can improve the best match, $\|f_{A'} - f_q\|_2 < \|f_B - f_q\|_2$, resulting in a good VPR localization estimate $p_q \approx p_{A'}$. The existence of this effect is demonstrated using constructed failure cases in the experiments of sub-section 6.3.7. In CtF localization, RPE afterward still reduces this gap further by estimating $\Delta p_{qA'} = L(f_q, f_{A'})$, such that $p_q = p_{A'} + \Delta p_{qA'}$. CoPR and RPE are therefore complementary techniques.

6.3 EXPERIMENTS

In this section, the experimental setup is presented in detail, including the datasets, baselines, and evaluation metrics. First, the author validates using the encoder $G_{distance}$ as the primary encoder. Then, the results of using descriptor regression for interpolation and extrapolation experiments are presented. The author shows how different feature encoders can benefit from CoPR and the effect of map density on localization performance. The relation between CoPR and CtF localization is shown, and finally, the computational details for this chapter are provided.

6.3.1 EXPERIMENTAL SETUP

Here, the datasets, evaluation metrics, and the various parametric choices used in the experiments are explained.

DATASETS

Three datasets are used for evaluation, Microsoft 7-scenes, the Synthetic Shop Facade, and the Station Escalator dataset. The choice of these datasets is based on their wide adoption for evaluating VBL in existing literature as reviewed previously, and their complementary nature: indoor vs outdoor, different levels of spatial coverage, and different types (parallel vs intersecting) of traversals. Each dataset is discussed in turn.

Microsoft 7-scenes dataset [242] has been a long-standing public benchmark for 6-DoF indoor localization [21, 29, 31]. This dataset consists of seven different indoor scenes

collected using a Kinect RGB-D camera and provides accurate 6-DoF ground-truth poses computed using a KinectFusion [245] baseline. Each scene spans an area of a few square meters and contains multiple sequences/traverses (viewpoint-varied) within a scene. Each sequence itself then contains between 500 to 1000 images, where each image has a 640×480 pixels resolution. There are separate query and reference sequences that contain novel viewpoints of the same scene. The images and poses in the query trajectory act as the training set for training both the feature encoder G and the non-linear descriptor regressor H. The reference trajectory is further divided into two splits: validation and test sets, with 40% images in the validation set and 60% images in the test set. The validation set is used for validating the encoder G and the non-linear regression network H at training time. This reference trajectory is then used for the interpolation and extrapolation experiments.

The **Synthetic Shop Facade dataset** proposed by [29] represents images and poses regressed from a 3D model of a real-world outdoor shopping street [19] and consists of multiple sequences/traverses of a single scene. It contains about 9500 images at novel viewpoints with an image resolution of 455×256 pixels. There are separate splits for query and reference sequences that contain different viewpoints. The training, validation, and test sets follow the same strategy as that of the 7-scenes dataset.

The **Station Escalator dataset** proposed by [29] contains two parallel trajectories through a station and is hence useful for studying extrapolation benefits across parallel lanes. The dataset contains 330 query images and 330 reference images with an image resolution of 1557×642 pixels and 6-DoF accurate poses. For this dataset, the author intends to regress descriptors from one trajectory (say A) to its parallel trajectory (say B), thus the non-linear regression network H needs to be trained with such relative pose change between A and B. Therefore, given the two original parallel trajectories, both are divided into three parts: training, validation, and test sets. The training images are selected as every 50th image in both trajectories, while the remaining images are equally divided between the validation and test sets. The training images from both traverses are used to train the descriptor regression models. For experiments, the validation and test images from trajectory A combined together act as the query images. The validation and test images from trajectory B in addition to the training images from trajectory A act as the reference images.

EVALUATION METRICS

The evaluation metric is the Median Translation Error (MTE) in meters and the Median Rotation Error (MRE) in degrees over all the estimated query images' poses, as commonly used in existing literature [21] [29] [31]. The median is normally preferred over the mean since outliers can skew the latter by any amount. The translation error is the Euclidean distance between the query image's translation and the best-matched reference image's translation. The rotation error is the angular difference between the quaternion vectors of a query image and its best-matched reference image, as used in the reviewed literature.

TRAINING DETAILS AND PARAMETRIC CHOICES

The output of the final global average pooling layer of a ResNet34 [246] backbone feature encoder is used, and thus the feature descriptor size is $N = 512$ throughout this chapter. The feature encoder G and the non-linear descriptor regressor H are trained separately. For training all the three feature encoders $G_{triplet}$, $G_{relative}$ and $G_{distance}$ and for non-linear regression network H, the Adam optimizer is used for model optimization with learning rates

of $1e^{-5}$, $1e^{-4}$, $5e^{-5}$ and $5e^{-4}$ for $G_{triplet}$, $G_{relative}$, $G_{distance}$ and H, respectively. The weights of the ResNet34 backbone are initialized via pretraining on ImageNet-1K and fine-tuned on the datasets used in this chapter, while the non-linear regression network H is trained from scratch for each dataset.

For training the encoder $G_{triplet}$, images from the training sets of different scenes of the 7-scenes dataset are chosen randomly to act as negatives, while images from the same scene with varied viewpoints are chosen as positives. The author uses a margin of $m = 0.3$ for the triplet loss, same as [80]. The feature encoder G is trained jointly on the training pairs of all the seven scenes in the 7-scenes dataset. The encoders trained using triplet loss ($G_{triplet}$) and RPE loss ($G_{relative}$) are only trained on the 7-scenes dataset and used for experiments on all the datasets, while the model trained using distance-based loss ($G_{distance}$) is trained separately for each dataset. The author later shows the reasons behind this separate training for distance-based loss in sub-section 6.3.2.

A dedicated non-linear regression model H is trained for each of the three datasets. The non-linear regression model H trained for one dataset is used for both the interpolation and extrapolation experiments of that dataset. For the least-squares plane fit to linearly regress each feature dimension, O=4 is chosen as the number of NN anchors, which is the minimum number needed to fit a plane in 4D (i.e., 3D world plus 1D feature).

6.3.2 ENCODER LOSS FUNCTION AND LOCALIZATION ACCURACY

Here, the author intends to understand the first part of the two potential requirements for accurate VPR-based localization: viewpoint variance. The encoder training objectives favouring viewpoint variance can have a considerable effect on the VPR-based localization error. The change in localization error for $G_{triplet}$, $G_{relative}$, and $G_{distance}$ is shown in Fig. 6.6 for the 7-scenes dataset, where a distance-based loss leads to the lowest localization error. This localization error is without map densification and is purely the effect of different training objectives for the encoder G.

Moreover, in Fig. 6.7, we can observe the (de)generalization of these feature encoders from one dataset to the other. This is done by evaluating the VPR-based localization performance of a given encoder on datasets other than the training dataset for a given model. The author notes that the network $G_{distance}$ trained on the 7-scenes dataset does not perform well on the Shop Facade dataset and is outperformed by $G_{triplet}$ and $G_{relative}$ trained on the 7-scenes dataset, which suggests that $G_{distance}$ is less generalizable. The author therefore trains $G_{distance}$ on the Shop Facade dataset, after which it outperforms the other networks. This degeneralization of distance-based loss has also been reported by [160], and an intuitive explanation could be that distance-based losses are more sensitive to structural changes between different domains and the change in scene appearance with changing scene depth.

Since distance-based loss leads to the lowest localization error, only $G_{distance}$ is used as the backbone encoder for the experiments in sub-sections 6.3.3 and 6.3.4. However, it is later shown in sub-section 6.3.5 that all the encoders ($G_{triplet}$, $G_{relative}$ and $G_{distance}$) can benefit from CoPR, albeit at varying levels of accuracy.

6.3.3 EXTRAPOLATION EXPERIMENTS

First, the setup used for the extrapolation experiments is explained, followed by the extrapolation methods and baselines, and then the corresponding results and discussion.

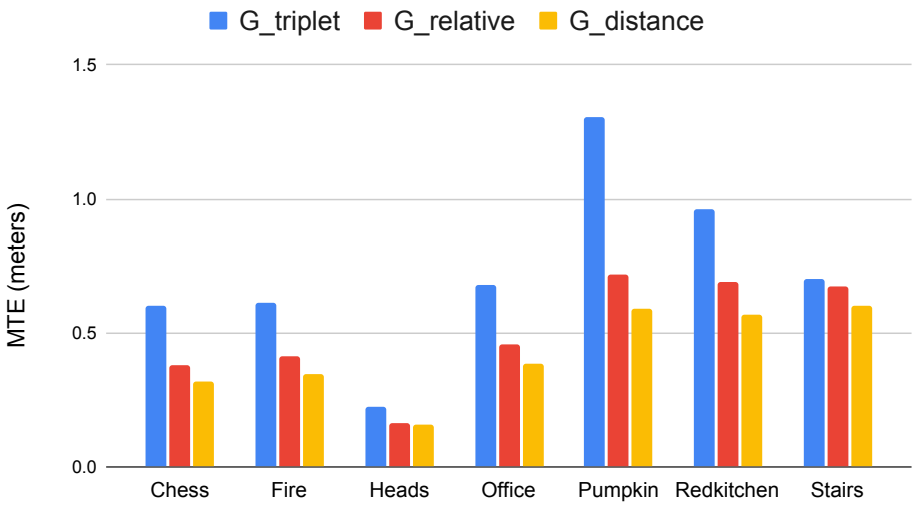


Figure 6.6: The MTE of the three encoders when used for performing VPR-based localization on all the scenes of the 7-scenes dataset. Training with distance-based loss leads to lower MTE than other losses.

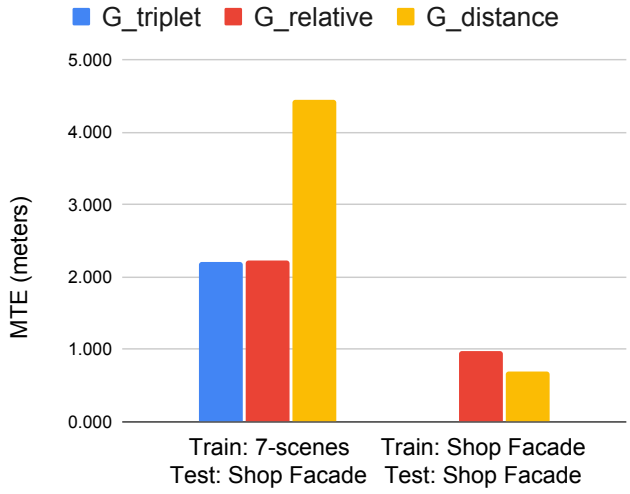


Figure 6.7: The MTE of the three encoders used for testing VPR-based localization on the Synthetic Shop Facade dataset, when trained on the same and different dataset. Notably, $G_{triplet}$ and $G_{relative}$ trained on the 7-scenes can outperform $G_{distance}$ trained on the 7-scenes dataset. However, $G_{distance}$, when trained and tested on the Synthetic Shop Facade dataset, performs the best. Since the Shop Facade dataset contains images of only one scene, unlike the 7-scenes dataset, the author could not select proper negative images in this dataset and did not train $G_{triplet}$ on this dataset.

EXTRAPOLATION SETUP

All three datasets are used to examine the effects of extrapolation. These three datasets have properties useful for our CoPR analysis. Thus, the author first explains the setup for extrapolation on these three datasets, as follows.

The extrapolation experiments are performed on all scenes of the **7-scenes dataset**. For each scene in the 7-scenes dataset, there are multiple reference sequences, thus, one of the reference traverses/sequences is taken as the anchor reference trajectory. The remaining reference sequences are then discarded[3] to get the original sparse map M_{sparse}. Then, on the selected reference sequence, the author selects every Kth sample (where $K = 50$) as the anchor point. Then, for each anchor point, target points are sampled uniformly in the x and y direction, keeping the viewing direction and z fixed to get the dense extrapolated map M_{dense}. The sampling of target points is done with a fixed step size e_{step} and a maximum spatial span e_{span} for extrapolation. A step size of $e_{step} = 0.05$ meters is used for all seven scenes, and the spatial span e_{span} is set to cover the complete area of the scene. Examples of this extrapolation are shown in Fig. 6.2 for the 7-scenes dataset.

The **Synthetic Shop Facade dataset** provides a query sequence, a single anchor reference sequence, and multiple target reference points sampled uniformly over a fixed grid across this anchor reference sequence. This already provided distinction is used to get M_{sparse} and M_{dense}. The query, anchor, and target extrapolated points contain novel viewpoints of the same scene, and the author refers the reader to the Sattler et al.'s [29] figure here[4] for visualization of the scene and target point distribution.

In the case of the **Station Escalator dataset**, the anchor reference images act as the sparse reference map M_{sparse}. Extrapolation on the Station Escalator dataset is straightforward: all images on the reference trajectory act as the anchor points, and a target descriptor is regressed using each anchor at an offset of 1.8 meters on the x-axis from the anchor reference pose. Then, the target descriptors combined with M_{sparse} descriptors act as the extrapolated map M_{dense}.

EXTRAPOLATION METHODS

Two descriptor regression methods are compared for extrapolation. **Linear Regression (Lin. Reg.)** is the local plane fit method introduced in sub-section 6.2.2. For the 7-scenes and the Shop Facade dataset, the O NN anchor points are selected from the reference trajectory, and for the Station Escalator dataset, two NN anchor points are selected from each of the two parallel trajectories A and B.

Non-linear Regression Network (Non-lin. Reg.) is the neural network regression approach from sub-section 6.2.2.

EXTRAPOLATION BASELINES

Sparse Map: The primary baseline for extrapolation is the sparse map M_{sparse}, where feature descriptors are only available at sparse poses P.

3D model: As mentioned in sub-section 6.3.1, the Shop Facade dataset already provides distinct anchor reference points and target extrapolation points. Since the images for these

[3] If we do not discard other reference sequences during extrapolation experiment, they overlap with target extrapolated/regressed descriptors and make the experimental setup less challenging.

[4] `https://github.com/tsattler/understanding_apr`

target extrapolation points are already available, their corresponding feature descriptors at all poses in the extrapolated map can also be computed. The author refers to this method as *3D Model* in the results, where the feature descriptors at all locations (anchor and non-anchor) in M_{dense} are computed using $G_{distance}$ and no descriptor is regressed. This baseline of [29] helps to understand how well the proposed extrapolation performs in comparison to having the ground-truth images at all locations in the extrapolated map.

Oracle retrieval: The author also shows the minimum possible translation error and the corresponding rotation error obtained by an oracle retrieval method, which always retrieves the ground-truth 3D Euclidean closest match in the extrapolated map M_{dense}. These errors indicate the VPR base errors for the used queries, and would only be zero if the query poses coincide with the reference poses in the map.

Table 6.1: The extrapolation experiments on the 7-scenes dataset. The MTE and MRE are reported. The oracle retrieval shows the minimum achievable MTE and the corresponding MRE. Best in bold.

Metric	Map	Densification	Retrieval	Chess	Fire	Heads	Office	Pumpkin	Redkitchen	Stairs	Avg.
MTE (m)	M_{dense}	-	*Oracle*	0.083	0.070	0.030	0.072	0.077	0.216	0.119	0.095
MTE (m)	M_{sparse}	-	VPR	0.318	0.348	**0.158**	0.383	0.589	0.567	0.600	0.423
MTE (m)	M_{dense}	*Lin. Reg.*	VPR	0.245	0.310	0.163	0.338	0.426	0.444	0.532	0.351
MTE (m)	M_{dense}	*Non-lin. Reg.*	VPR	**0.167**	**0.279**	0.159	**0.264**	**0.346**	**0.427**	**0.430**	**0.296**
MRE (°)	M_{dense}	-	*Oracle*	28.44	25.56	21.25	58.37	56.33	35.97	23.85	35.68
MRE (°)	M_{sparse}	-	VPR	**22.54**	20.88	**16.49**	**38.89**	**44.89**	34.65	24.32	**28.95**
MRE (°)	M_{dense}	*Lin. Reg.*	VPR	29.04	**18.49**	16.62	39.10	61.96	**33.00**	25.41	31.95
MRE (°)	M_{dense}	*Non-lin. Reg.*	VPR	26.87	22.02	16.54	47.95	58.90	36.33	**21.29**	32.84

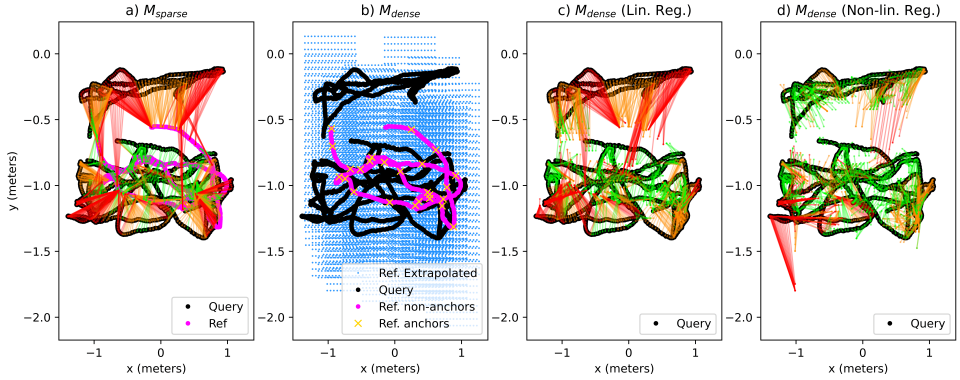


Figure 6.8: Extrapolation experiments on the Office scene of the 7-scenes dataset. (a) The matches between the query and the reference points for the sparse map M_{sparse}, (b) the poses in the densified map M_{dense}, (c) the matches in map densified using *Lin. Reg.*, (d) the matches in map densified using *Non-lin. Reg.*. All matches are color-coded as *green*, *orange*, and *red* with increasing 3D Euclidean distance in the physical space. The reference poses in (c) and (d) are the same as in (b) and thus are not shown to avoid cluttering. The non-linearly densified map (d) clearly leads to better performance than other maps, albeit with some failure cases towards the bottom-left of the plot.

EXTRAPOLATION RESULTS

The extrapolation results are reported in Table 6.1 for the originally sparse, linearly extrapolated, and non-linearly extrapolated maps for all the seven scenes in the 7-scenes

dataset. The matches between the query and the reference trajectories for the extrapolation experiment are shown in Fig. 6.8 for the Stairs scene of the 7-scenes dataset as an example. It can be seen that extrapolation leads to significant performance improvement over no extrapolation in terms of translation error. By using extrapolation, the descriptors closer to the query trajectory are matched. The author also notes that the non-linear regression model H performs better than the linear regression model, indicating that extrapolating across the trajectory requires a non-linear approach to handle the complexity of the feature space. Performance improvement is not observed in the translation error due to extrapolation on the Heads scene, where the query and the reference trajectories are already relatively close to each other compared to the other scenes. Moreover, it is observed that with the current map densification setup, the angular estimation cannot be improved. However, it is important to notice that even retrieving the Euclidean closest match in physical space leads to an increase in rotation error, as shown by *Oracle* retrieval in Tables 6.1 and 6.2. This increase in rotation error and the reasons behind it are further discussed in Section 6.4.

The same findings are extended to the Synthetic Shop Facade dataset as reported in Table 6.2. We can see performance improvement thanks to extrapolation, and the non-linear regression model H outperforms linear regression. It also observed that the VPR performance of the non-linearly extrapolated map (*Non-lin. Reg.*) is similar to the map densified using 3D modelling, which suggests that the trained non-linear regression model H closely regresses the original descriptors, without access to the images at the target poses.

The results on the Station Escalator dataset also support the motivation of this chapter, since the localization accuracy is significantly improved, as reported in Table 6.2. The author also shows the qualitative results on the Station Escalator dataset in Fig. 6.9. These results highlight the utility of descriptor regression in cases where parallel traverses are common, such as highway lanes, train tracks, escalators, and many such laterally viewpoint-varied paths.

More benefits of non-linear descriptor regression are observed on the Station Escalator dataset than on other datasets. Linear regression does not work well on this dataset; the selected anchor poses are too distant from the query trajectory. Recall that for this dataset, the training pairs include sparse samples (every K-th image) from both the query and reference traverses to increase the variance in the training data, as there are only two traverses in total in this dataset. Still, the extrapolation experiments do not extrapolate to the exact query locations but to nearby locations. The author observes that training with similar relative pose differences as those observed at test time leads to performance benefits. In a real-world application, if only sparsely sampled images are collected for parallel trajectories, the pose differences are representative for training a regression model and densifying the trajectories for improved localization accuracy.

6.3.4 INTERPOLATION EXPERIMENTS

Now the setup used for interpolation experiments is explained, followed by the methods and baselines, and then the corresponding results and discussion.

INTERPOLATION SETUP

The interpolation experiments are performed on all the scenes in the 7-scenes dataset. Similar to the extrapolation setup, interpolation uses the same concept of a sparse map

Table 6.2: The extrapolation experiments on the Synthetic Shop Facade and the Station Escalator datasets. The MTE and MRE are reported. The oracle retrieval shows the minimum achievable MTE and the corresponding MRE. Best in bold.

Metric	Map	Densification	Retrieval	Shop Facade	Station Escalator
MTE (m)	M_{dense}	-	*Oracle*	0.188	0.26
MTE (m)	M_{sparse}	-	VPR	0.705	2.17
MTE (m)	M_{dense}	*3D Model* [29]	VPR	**0.335**	NA
MTE (m)	M_{dense}	*Lin. Reg.*	VPR	0.541	2.10
MTE (m)	M_{dense}	*Non-lin. Reg.*	VPR	0.344	**0.94**
MRE (°)	M_{dense}	-	*Oracle*	11.25	9.45
MRE (°)	M_{sparse}	-	VPR	**10.99**	**8.54**
MRE (°)	M_{dense}	*3D Model* [29]	VPR	11.13	NA
MRE (°)	M_{dense}	*Lin. Reg.*	VPR	**10.99**	8.60
MRE (°)	M_{dense}	*Non-lin. Reg.*	VPR	11.13	8.99

M_{sparse} and a dense map M_{dense}, though for the interpolation experiments, these maps are defined differently than for the extrapolation experiments. For interpolation, the full reference trajectory of a scene is used as the *ground-truth* dense map M_{dense}. The author then sub-samples the reference trajectories by a factor of $K = 50$, such that the consecutive images in a trajectory still contain visual content overlap. This reduced set of references is used as the sparse map M_{sparse}. The *ground-truth* dense map serves as a baseline that can assess the performance of VPR if densely sampled reference images would be available, while the sub-sampled version shows the performance when only a sparse set of reference images are available. Examples of this sub-sampling are shown in Fig. 6.2. For CoPR, the poses in P from the sparse map act as the anchor poses, while the additional poses P' found in the ground-truth dense map act as the target poses. All feature descriptors in M_{sparse} and the query descriptors are computed using the feature encoder $G_{distance}$ explained in sub-section 6.2.3.

INTERPOLATION METHODS

The compared descriptor regression methods are the simple **Linear Interpolation (Lin. Interp.)** from sub-section 6.2.2; the **Linear Regression (Lin. Reg.,)** from sub-section 6.2.2; and the **Non-linear Regression Network (Non-lin. Reg.)** from sub-section 6.2.2.

INTERPOLATION BASELINES

Sparse map: The primary baseline for interpolation is the sparse map M_{sparse}, where feature descriptors are only available at sparse poses P.

Ground-truth dense map: Unlike the extrapolation experiments, where there are no true images (and hence descriptors) available at target poses, in the case of interpolation experiments, these true images are available. Thus, this *Ground-Truth* (GT) dense map M_{dense} is a baseline that serves the true descriptors for the target poses.

Oracle retrieval: The author also shows again the minimum possible translation error and the corresponding rotation error from the oracle retrieval method, as defined in sub-section 6.3.3.

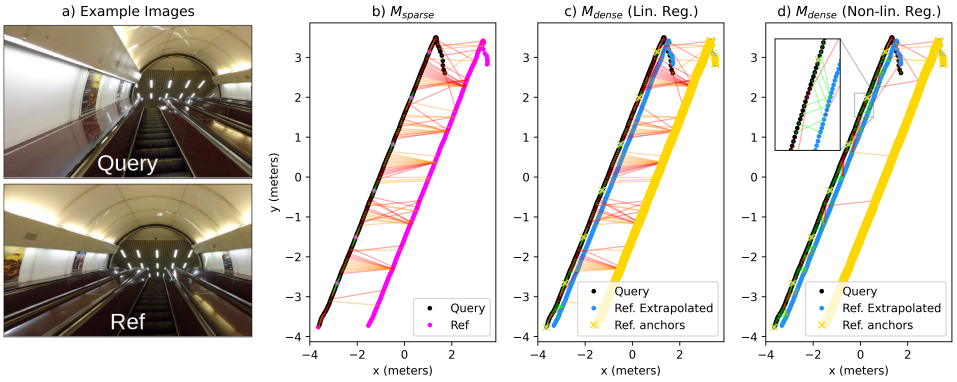


Figure 6.9: Extrapolation experiments on the Station Escalator dataset. (a) Exemplar query and reference images. Then the matches between the query and the reference points for the (b) original sparse map M_{sparse}, (c) linearly regressed (*Lin. Reg.*) map M_{dense} and (d) non-linearly regressed (*Non-lin. Reg.*) map M_{dense}. These matches are color-coded as *green*, *orange*, and *red* with increasing 3D Euclidean distance in the physical space. Extrapolation with non-linear regression network H is done using only the points on the anchor reference trajectory in *yellow* on the right, whereas the sparse anchor points in *yellow* on the query trajectory are only used at training time.

6

INTERPOLATION RESULTS

The results for all the methods and baselines for the interpolation experiment on the 7-scenes dataset are reported in Table 6.3 for all the seven scenes. The VPR matches between the query and reference trajectories for the Heads scene are shown in Fig. 6.10. A general decrease in localization error can be seen when moving from the sparse map M_{sparse} to the *GT* dense map M_{dense}. Interestingly, it can also be seen that even simple linear regression (*Lin. Reg.* and *Lin. Interp.*) can solve this problem well and is often the best-performing technique. Note though that linear regression is done using multiple anchor points, which constraints the problem setting, while the non-linear regression network H only uses one anchor point. Nevertheless, this experiment shows that map densification even via interpolating along the trajectory is helpful, although it has lesser benefits than extrapolation across the trajectory.

The observed differences between the interpolation and extrapolation experiments are discussed in more detail in the Discussion, Section 6.4.

6.3.5 MAP DENSIFICATION WITH DIFFERENT FEATURE ENCODERS

Next, the author tests that using the non-linear regression model H for extrapolating across anchor points is beneficial for all discussed feature encoders. This is reported in Table 6.4. However, the corresponding localization accuracy is limited by the localization performance of the respective feature encoder. The MTE is reported for all three types of feature encoders on the sparse map M_{sparse} and the non-linearly regressed (*Non-lin. Reg.*) map M_{dense} for the 7-scenes dataset and the Synthetic Shop Facade dataset. Such a generic boost of performance using map densification supports that CoPR can utilize inherent benefits of different types of feature encoders, for example, the domain generalization of $G_{triplet}$ and $G_{relative}$, and the viewpoint variance of $G_{distance}$.

Table 6.3: The interpolation experiments for the 7-scenes dataset at K=50. The MTE and MRE are reported. The oracle retrieval shows the minimum achievable MTE and the corresponding MRE. Best is in Bold.

Metric	Map	Densification	Retrieval	Chess	Fire	Heads	Office	Pumpkin	Redkitchen	Stairs	Avg.
MTE (m)	M_{dense}	-	*Oracle*	0.109	0.183	0.097	0.117	0.115	0.129	0.132	0.126
MTE (m)	M_{dense}	*GT Map*	VPR	0.165	0.255	0.158	0.207	0.242	0.219	0.261	0.215
MTE (m)	M_{sparse}	-	VPR	0.210	0.322	0.212	0.237	0.250	0.271	0.263	0.252
MTE (m)	M_{dense}	*Lin. Interp.*	VPR	0.170	0.277	0.202	**0.211**	0.257	**0.220**	**0.257**	0.227
MTE (m)	M_{dense}	*Lin. Reg.*	VPR	**0.169**	**0.257**	**0.165**	0.216	**0.214**	0.224	0.262	**0.215**
MTE (m)	M_{dense}	*Non-lin. Reg.*	VPR	0.178	0.264	0.184	0.221	0.259	0.260	0.278	0.234
MRE (°)	M_{dense}	-	*Oracle*	22.81	26.65	20.91	43.56	37.58	31.31	29.71	30.36
MRE (°)	M_{dense}	*GT Map*	VPR	17.69	19.71	16.49	32.13	36.24	22.49	19.55	23.47
MRE (°)	M_{sparse}	-	VPR	20.75	**19.15**	19.34	35.01	**33.96**	27.27	**19.16**	24.94
MRE (°)	M_{dense}	*Lin. Interp.*	VPR	21.73	20.65	17.93	**34.65**	39.15	26.27	19.92	25.75
MRE (°)	M_{dense}	*Lin. Reg.*	VPR	**20.09**	20.00	**17.20**	35.63	34.00	**24.16**	19.72	**24.40**
MRE (°)	M_{dense}	*Non-lin. Reg.*	VPR	22.09	19.45	20.13	39.73	39.03	27.17	20.25	26.83

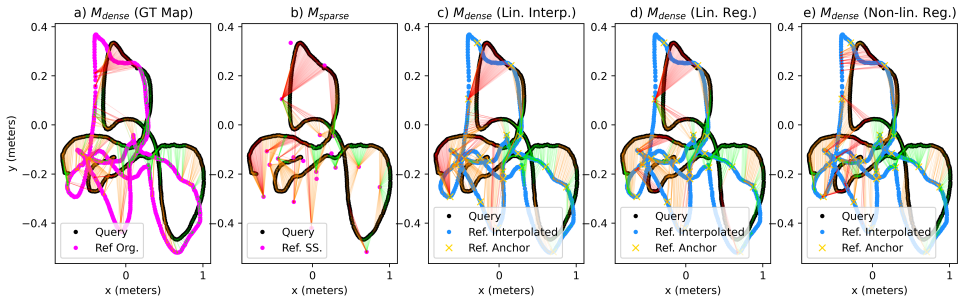


Figure 6.10: Interpolation experiments on the Heads scene of the 7-scenes dataset. The matches between the query and the reference trajectories in (a) the *GT* dense map M_{dense}, (b) the sparse map M_{sparse}, (c) the linearly regressed (*Lin. Reg.*) map M_{dense}, (d) the linearly interpolated (*Lin. Interp.*) map M_{dense} and (e) the non-linearly regressed (*Non-lin. Reg.*) map M_{dense} given K=50. The matches are color-coded as *green*, *orange*, and *red* with increasing 3D Euclidean distance in the physical space.

Table 6.4: The effect of CoPR on different feature encoders on all scenes from the 7-scenes dataset, and on the Synthetic Shop Facade dataset. In the case of $G_{triplet}$, the author uses the triplet loss as motivated by the authors [80] but does not use the VLAD descriptor module to keep the backbone the same for a fair comparison across all encoders.

Feature Encoder	$G_{triplet}$		$G_{relative}$		$G_{distance}$	
Reference Map	M_{sparse}	M_{dense}	M_{sparse}	M_{dense}	M_{sparse}	$Mdense$
7-scenes - Chess	0.600	**0.450**	0.379	**0.260**	0.318	**0.167**
7-scenes - Fire	0.612	**0.542**	0.414	**0.296**	0.348	**0.279**
7-scenes - Heads	0.227	**0.215**	0.166	**0.147**	**0.158**	0.159
7-scenes - Office	0.680	**0.589**	0.455	**0.246**	0.383	**0.264**
7-scenes - Pumpkin	1.306	**1.208**	0.720	**0.479**	0.589	**0.346**
7-scenes - Redkitchen	0.960	**0.783**	0.691	**0.451**	0.567	**0.427**
7-scenes - Stairs	**0.699**	0.780	0.673	**0.374**	0.600	**0.430**
Synthetic Shop Facade	2.234	**1.419**	2.219	**1.641**	0.705	**0.344**
Average	0.915	**0.748**	0.715	**0.487**	0.458	**0.302**

6.3.6 MAP-DENSITY VS LOCALIZATION ACCURACY

The motivation presented in this chapter suggests that the denser the reference map, the lower the localization error of a VPR-based localization system. In this chapter, this map density is modelled with the step size e_{step}. Therefore, in this sub-section, the author shows the effect of increasing map density on the localization error by using extrapolation with non-linear regression model H and feature encoder $G_{distance}$ for the 7-scenes dataset. This direct relation between the step size e_{step} and the MTE is presented in Fig. 6.11. Decreasing the step size leads to denser extrapolated maps which then leads to a decrease in MTE for the non-linearly extrapolated (*Non. Lin. Reg.*) map M_{dense}. The performance benefits for the scenes depend on the underlying scene geometry and the quality of descriptor regression. For example, in the case of Heads scene, the query poses and the sparse reference poses in M_{sparse} are already close to each other, thus we cannot see any performance benefits due to densification. While in other scenes, we can see that map densification is helpful and is related to the level of map densification modelled with the step size e_{step}.

6.3.7 BENEFITS OF COPR FOR RPE

In this section, we will look into the relation of CoPR with RPE and hence CtF localization, as discussed in sub-section 6.2.4. In this experiment, the author makes this argument concrete by illustrating that situations exist where a sparse map leads to incorrect coarse retrieval of a visually similar image descriptor taken at an arbitrarily far location, which can in turn lead to the failure of CtF approaches. It is argued that this error source is fundamentally due to the retrieval step, not due to the subsequent RPE step. Furthermore, it is demonstrated that map densification could tackle this error source in some cases.

Exemplar cases are created in the 7-scenes dataset where such an effect can be easily observed. A reference database of four sparsely sampled reference images around a query image is created for a given scene, and a fifth *stray* reference image is added to this reference database. This stray image is taken from a completely different scene that has no real physical overlap with the query image. The author uses the feature encoder $G_{relative}$ for

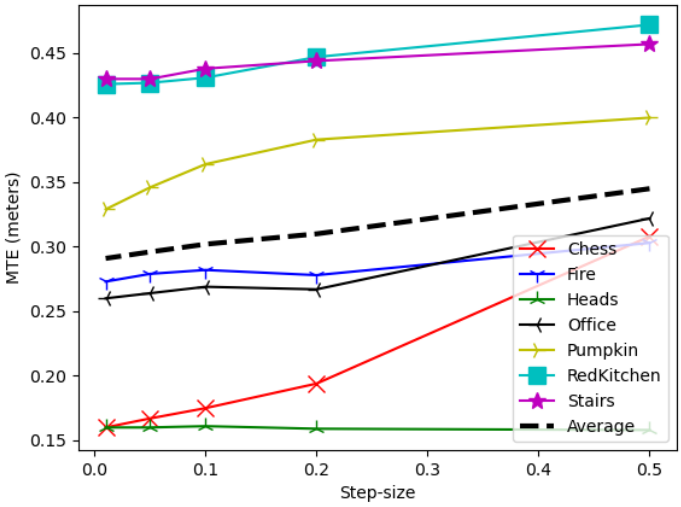


Figure 6.11: The increase in MTE by increasing the step size e_{step} for all scenes of the 7-scenes dataset. A larger step size leads to sparser maps which increases the translation error, while a smaller step size leads to denser maps, which are useful for accurate localization.

image-retrieval and the non-linear descriptor regression network H to regress the expected descriptor at the query location given the nearest anchor reference descriptor. This regressed descriptor acts as the descriptor for a hypothetical 6th image in the reference database at the query location.

The objective of this experiment is to show that in the absence of the regressed descriptor, the stray image is selected as the best match for the query image, while in the presence of the regressed descriptor, the stray image is pushed downwards in the list of retrieved images ranked by their matching scores. Please note that in the case where the stray image is chosen as the best match, the localization error can be arbitrarily large, as a different scene can be quite far. The author shows four such example cases in Fig. 6.12 from the 7-scenes dataset, where we can observe that in the absence of the regressed descriptor, the stray image is chosen as the best match by the image-retrieval system. Since such stray cases are shown to exist in multiple scenes of the 7-scenes dataset, which is a small-scale dataset, this effect would amplify even further in spatially larger scenes due to the increased chances of perceptual aliasing.

Thus, without CoPR, sparse reference maps *could* lead to incorrect coarse retrieval, where the coarse pose estimate can be arbitrarily far-away and hence cannot be corrected by CtF approaches. By using CoPR, reference descriptors of the correct scene now appear close to the query descriptor. Finding all references near the query in the feature space thus identifies similar scenes, allowing to at least represent localization ambiguity and ideally obtaining a correct best match. Without CoPR, only the incorrect scene would have matched

the query. Better retrieval also benefits CtF approaches, since the RPE step is only valid if the retrieved reference pose represents the correct scene. These constructed cases illustrate that CoPR and CtF are complementary approaches to improve VPR-based localization accuracy. Please note that this analysis does not demonstrate that CoPR prevents false positives as a general rule, but that it is possible to construct cases where the complementarity of CoPR and CtF can be observed. Future work may investigate this further.

6.3.8 COMPUTATIONAL DETAILS

Finally, the sizes of the sparse and dense maps, the time spent t_{dense} on creating the dense maps M_{dense} using H, and the training times t_{train} of model H are reported for all the datasets in Table 6.5. For the 7-scenes dataset, the results are reported for the Office scene. The retrieval time t_{retr} in VPR is the sum of the time t_{enc} required to encode a query image into a feature descriptor and the time t_{match} spent to find the NN match of this descriptor in the map. Since the encoding time is several times higher than the efficient NN search, the retrieval time is not too affected by map densification. Please note that the timings are not comparable between the datasets due to differences in map content (i.e., descriptors).

Table 6.5: The computational footprint of CoPR, please see accompanying text for details.

	Map	**7-scenes**	**Shop Fac.**	**Stat. Esc.**
$t_{train}(sec)$	-	510	540	960
$t_{dense}(msec)$	-	12.8	2.241	0.32
$t_{enc}(msec)$	-	6.16	8.39	5.88
$t_{match}(msec)$	M_{sparse}	0.02	0.1	0.02
	M_{dense}	0.05	0.32	0.08
$t_{retr}(msec)$	M_{sparse}	6.18	8.49	5.90
	M_{dense}	6.21	8.71	5.96
Map Size (#)	M_{sparse}	1000	231	337
	M_{dense}	13000	2531	667

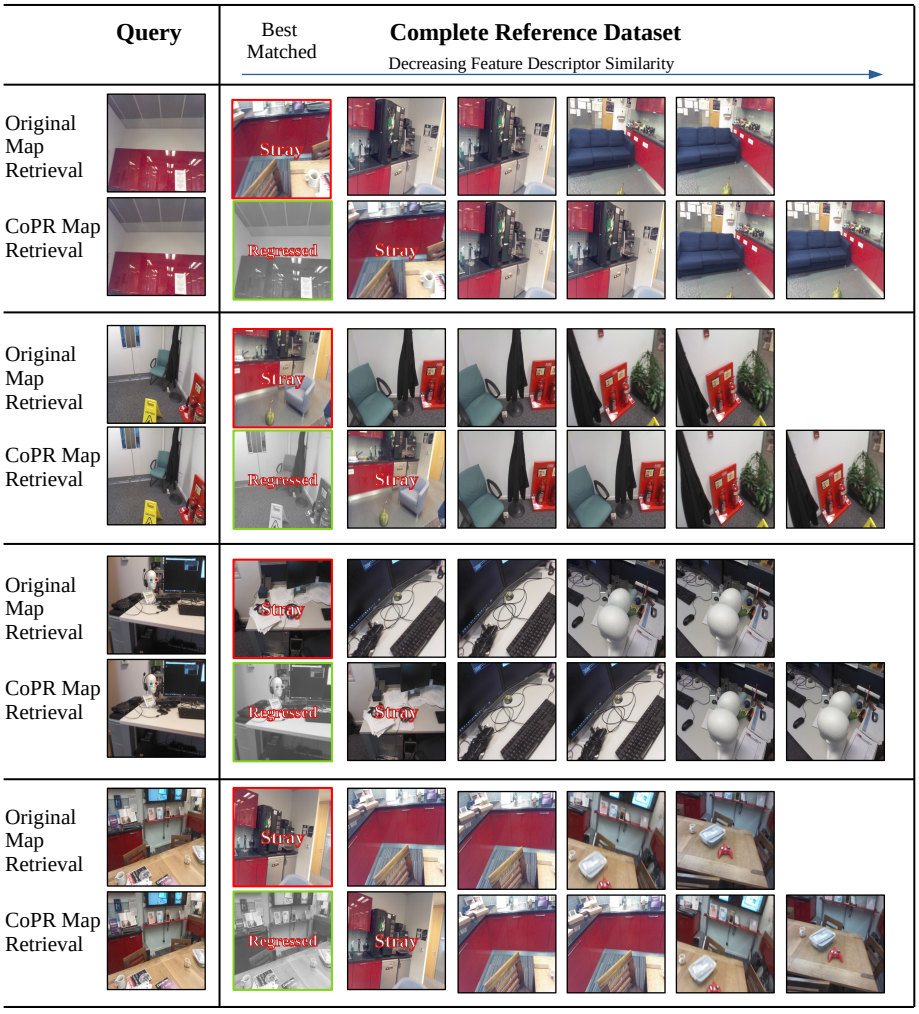


Figure 6.12: The exemplar cases where image-retrieval fails to retrieve useful coarse estimates for RPE in a sparse reference map. By regressing the expected descriptor at the query pose, the author shows that map densification could lead to robustness against such failure cases. The grayscale image in the reference set is only added for the reader's reference and represents only a hypothetical image for the regressed descriptor at the query pose, since we do not synthesize images but only regress image descriptors. The *green* bounding box represents a correct match and the *red* bounding box represents an incorrect match.

6.4 DISCUSSION

In this section, the major limitations of the work in this chapter and areas that need further investigation are reported.

Angular error: In both the interpolation and extrapolation experiments, it is clear that the proposed approach does not improve angular localization accuracy, as reported in Tables 6.1, 6.2, and 6.3. However, it is also important to note that retrieving the ground-truth Euclidean closest match in the physical space also leads to an *increase* in angular error (MRE). This is because the nearest match in terms of translation may not have the same 3D orientation. Thus, we can attribute the increase in rotation error using CoPR to two reasons: firstly, during interpolation and extrapolation experiments, the author does not change the angular pose but only the translation pose, given the anchor points, for the target points, and secondly, the encoder $G_{distance}$ does not optimize for angular localization error in its training objective. Thus, reducing both the translation and angular error requires that the Euclidean closest match in the physical space has the closest angular orientation to the query image. Future works could look into the benefits of using distance+orientation based encoder loss along with map densification in a 6-DoF setting.

Ground-truth closest matches: The results on extrapolation show that map densification can lead to a significant decrease in localization error. Moreover, the extrapolation experiments on the Shop Facade dataset also show that the localization performance on the non-linearly extrapolated (Non-lin. Reg.) map M_{dense} is close to the localization performance on a ground-truth (obtained using 3D modelling) dense map M_{dense}. However, the localization error given the encoder $G_{distance}$ and the non-linear regression network H is still higher than the minimum possible localization error. The minimum possible translation error (*Oracle Retrieval*) in M_{dense} has been reported in Tables 6.1, 6.2 and 6.3. The author further shows qualitatively in Fig. 6.13, the performance that could be achieved by an oracle VPR system that always retrieves the Euclidean closest match in the physical space as the best match in a dense map. This gap in performance presents room for future research in this area. Furthermore, the results only show the generalization of non-linear data-driven regression model H across viewpoints within the same scene, however, generalization across scenes could be the new frontier for CoPR.

Interpolation vs extrapolation: From the results of the two experiments, it can be noted that the absolute decrease in localization error from interpolation is less than the decrease in localization error from extrapolation. The author hypothesizes two reasons for this: 1) the query trajectory has larger relative pose distance to the extrapolated poses than to the interpolated poses, 2) the viewpoint variance vs invariance of VPR encoders (as explained in sub-section 6.2.3) acts as a bottleneck, since the VPR system does not necessarily match the ground-truth Euclidean closest match in the physical space, but to *one of the closest* matches. The author expects that major performance benefits, given these experiments, require models that have even better viewpoint variance than the feature encoder $G_{distance}$. This motivates viewpoint-variant VPR for high accuracy, in addition to the existing trends for viewpoint-invariant VPR [223].

Generally, the author found that extrapolation is more useful than interpolation when a repeated traversal could occur at a laterally offset-ed path. Such trajectories are common to observe in real-world, for example, parallel traverses in outdoor scenes (Shop Facade dataset) and parallel traverses in indoor scenes (Station Escalator dataset). Other examples include

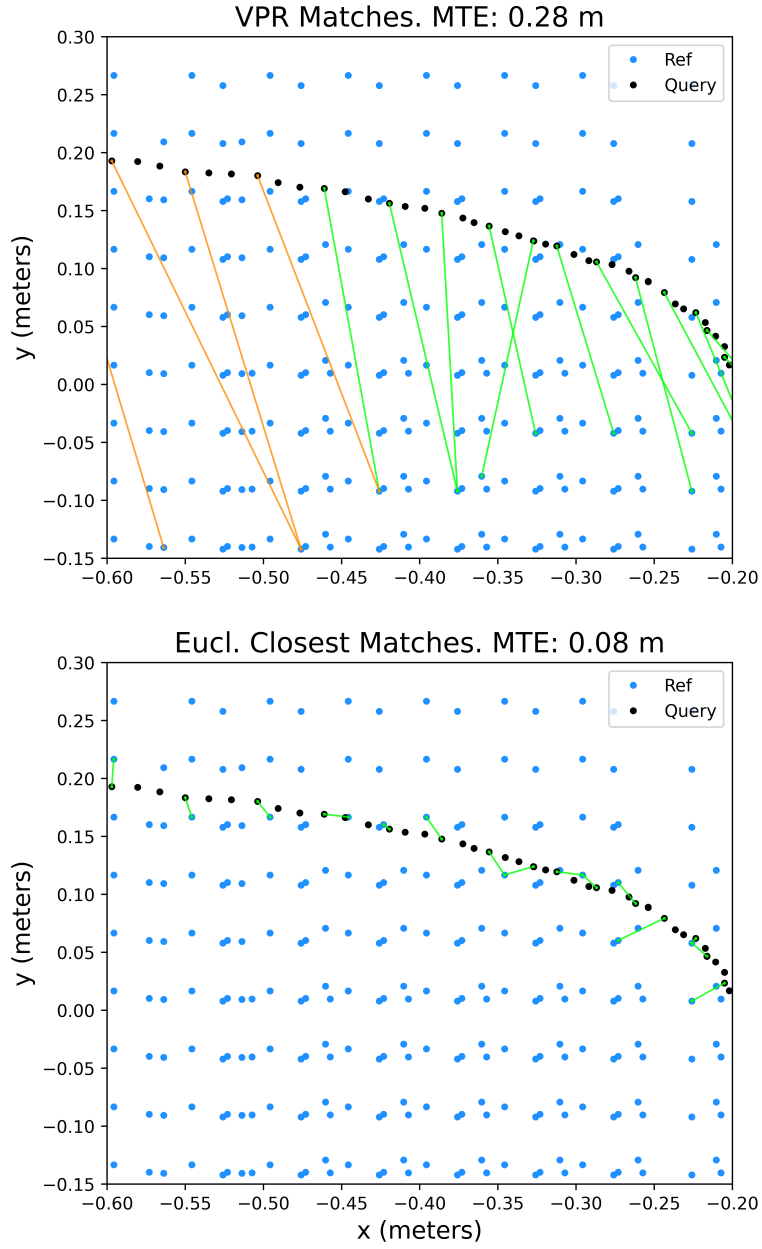


Figure 6.13: The VPR matches (top) and ground-truth 3D Euclidean closest matches in the physical space (bottom) between the query and the reference trajectories in the Fire scene of the 7-scenes dataset for the non-linearly extrapolated (*Non-lin. Reg.*) map M_{dense}. The matches are color-coded as *orange* and *green* with increasing 3D Euclidean distance. Although non-linearly regressed target poses (in *blue*) are matched to by VPR, these are not always the Euclidean closest matches in the physical space. Hence, there is still room for improvement.

lanes on a highway and parallel paths in corridors. However, the results in this chapter do show that both interpolating and extrapolating descriptors generally give better localization accuracy than using sparser reference maps, which suggests that map densification (CoPR) along the trajectory and/or across the anchor points can be useful for VPR.

6.5 CONCLUSIONS OF THE CHAPTER

In this chapter, the discrete treatment of places in a VPR map is investigated. The author has shown that map densification whether using interpolation or extrapolation is helpful to reduce translation error. The results for the 7-scenes dataset suggest that interpolating along the trajectory is an easier problem and can be solved with simple linear regression in the local neighborhood, however, extrapolation benefits from a non-linear treatment. Moreover, the proposed non-linear regression network only uses a single anchor point for regression, while the linear regression method uses multiple anchor points. The author validated that map densification is helpful for feature encoders trained with the three different types of losses, and that the highest accuracy is achieved when using a distance-based loss. Moreover, the benefit of map densification is shown for three datasets: 7-scenes, Synthetic Shop Facade, and Station Escalator, where each of them represents a different type of problem setting. It is also discussed that RPE and CoPR address related but complementary problems. Through several constructed cases, it is demonstrated that in a sparse map, localization might fail due to perceptual aliasing. RPE cannot recover the true location from a retrieved wrong place. CoPR helps retrieve the correct place, thus solving errors that RPE cannot.

While the distance-based loss function helps to retain viewpoint information among descriptors, it is observed that there is still room for improvement in comparison to retrieving the ground-truth Euclidean closest reference descriptors in the physical space. Future work could investigate architectures and loss functions that further enforce the network to learn feature representations useful for retrieving the 3D Euclidean closest match. As shown in this chapter, anchor selection and descriptor extrapolation are two separate steps for map densification. In the future, a separate treatment of both, i.e., learning good anchors and extrapolating well using multiple anchors, could lead to better map densification.

7

CONCLUSIONS

This chapter concludes this thesis by first presenting the key findings and answering the research questions. It then provides a broader discussion given the preceeding chapters. Finally, the author presents the future directions and his wishful thinking.

7.1 KEY FINDINGS

The key findings of this thesis are presented here. Primarily, this thesis addressed the open challenges in VPR using the cues available freely in the test-time reference map; however, VPR is still not fully solved in all three attributes: generalizability, uncertainty estimation, and localization accuracy.

No universal SOTA VPR method exists. Chapter 3 compared a large number of handcrafted and deep-learning-based VPR methods on many datasets and concludes that a single VPR method does not outperform on all datasets. In fact, for small-scale applications, such as household cleaning robots, basic handcrafted VPR techniques could work equally well as deep-learning-based methods. Chapter 4 further reinforces this by presenting results on more recent VPR techniques, and showcasing that the SOTA changes between datasets.

Changing the evaluation metrics changes the SOTA. VPR, as identified in Chapter 3, has been researched by two different communities: computer vision and robotics. Both of these communities motivate their evaluation metrics for different applications. Chapter 3 demonstrated that for the same dataset, a change of evaluation metric changes the SOTA. Thus, the choice of evaluation metrics is important in VPR and should be tied closely with the application of interest. Moreover, Chapter 6 presents that different training strategies are needed for different applications. A VPR backbone trained with a distance-based loss would perform well on distance-based metrics (Median Translation Error (MTE), Recall@0.5m, etc.) and is then useful in viewpoint-variant applications, such as localization, but not necessarily for viewpoint-invariant image retrieval.

Finetuning vision-foundation-model-based VPR methods on large datasets does not solve the train-test domain gap. Chapter 4 evaluated a number of SOTA VPR techniques

which use a vision foundation model backbone (DinoV2) and are trained on hundreds of thousands of images from diverse locations in Google Street View, revealing that the train-test domain gap is still a challenge for VPR. This suggests that the path towards a universal VPR method does not necessarily lie in the trend of employing larger models and/or larger training datasets.

The test-time reference set contains free information for improving both the generalization and uncertainty estimation of VPR methods. Chapter 4 demonstrated that several hundred or a few thousand images in the test-time reference map could introduce domain knowledge into VPR methods. A small learning rate combined with Kornia augmentations on the test-time reference map proved to be a simple recipe to improve the performance of SOTA VPR methods when tested in domains different from the training dataset.

Moreover, the test-time reference map is also useful for uncertainty estimation in VPR, especially when the reference map is densely collected. Chapter 5 presented Spatial-Uncertainty-Estimation (SUE) as a proxy for uncertainty in VPR and reported that it outperforms various heuristic and deep-learning-based uncertainty estimation methods. SUE was also demonstrated to be complementary to other forms of uncertainty estimation, e.g., the computationally-expensive geometric verification.

Uncertainty in VPR is a property of the test dataset. Chapter 5 discussed how uncertainty in VPR cannot generalize from a training dataset, since what should be considered as perceptually-aliased (uncertain) may differ between the training and test datasets. For example, images containing trees may always be considered uncertain if uncertainty is learned from a large training dataset, but if at test-time there is only one unique tree in the test dataset, such a retrieval *should* be considered certain. This is in contrast to the trend of learning uncertainty in VPR from training datasets [117, 118].

If the reference map is dense, VPR may be sufficient for accurate localization. Existing works in VPR have treated it as only a coarse localization method; however, Chapter 6 argues that if the reference trajectory is close to the query trajectory, VPR can provide sub-meter localization as a stand-alone module. Depending on the level of localization accuracy required, it is possible that a secondary computationally-expensive fine localization module (e.g., local-features-based relative pose estimation) is not needed.

Feature descriptors in VPR can be regressed at novel views. Continuous Place-Descriptor Regression (CoPR) is demonstrated in Chapter 6 where a non-linear neural network is trained to regress VPR descriptor at a novel view given an anchor descriptor and relative pose. While methods such as Neural-Radiance-Fields and Gaussian Splatting can regress images at novel views, CoPR does not operate on the image level, aligning with the motivation for VPR: high-dimensional images are not assumed available in the map and are replaced by robust feature vectors to save storage and computational costs.

7.2 ANSWERING THE RESEARCH QUESTIONS

The research questions for this thesis, as outlined in section 1, are answered here. The author first answers the sub-research questions to build up to the answer for the main question.

SQ1: Are the existing methods in VPR already generalizable, accurate, and uncertainty-aware?

Answer: Chapter 3 reported that there is no universal SOTA VPR method, and that different VPR methods can be the SOTA in changing test environments. This is the first indication of the lack of generalizability of different VPR methods, which is further demonstrated in Chapter 4 by reporting the declining VPR performance with increasing train-test domain gap. Not only do existing VPR methods lack generalization, but they are also accompanied by poor uncertainty estimation. Metrics like Precision-Recall curves in Chapters 3 and 5 demonstrate that existing uncertainty estimates fail to fully distinguish between true-positives and false-positives. Moreover, from Chapter 6 it is clear that VPR methods cannot be directly used for localization in sparse maps without a secondary relative pose estimation module.

SQ2: How can the test-time reference map be used to bridge the train-test domain gap in VPR?

Answer: The test-time reference map is generally assumed to be available offline in VPR. Although it does not fully represent the scene appearance expected at query time, e.g., the queries and references could be from entirely different seasons or times of day, the reference map still contains significant information (indoor/outdoor, urban/rural, etc.) about the test environment. This information is exploited in Chapter 4 by augmenting the reference images using Kornia [230] library to create pseudo-queries, then mining triplets using reference poses, and finally finetuning SOTA VPR methods on these triplets with a small learning rate. This strategy, namely Reference-Set-Finetuning (RSF) is shown to improve generalization of SOTA VPR methods to scenes different from the training datasets, such as train tracks (the Nordland dataset [191]) and archival imagery (the Amstertime dataset [228]).

SQ3: Could the poses in the test-time reference map and the spatial spread of the best-matching reference images help estimate uncertainty in VPR?

Answer: Uncertainty in VPR is a property of the test environment and cannot be generalized from the training dataset. The test-time reference map contains poses in many applications of VPR. Chapter 5 exploits these poses of the reference images to compute a spatial spread of the best retrieved references, and uses the variance of the pose distribution as a proxy for image matching uncertainty. If the top-retrieved references have high image matching scores (low distances in the feature-space) but are geographically far-apart, this indicates that the content in the query image is highly similar across different locations in the map, i.e., perceptually aliased. On the other hand, if the top-retrieved references are concentrated in a small geographical radius, the variance of the retrieved reference poses distribution will be small, indicating high confidence in the retrieval. The contribution of reference poses to the variance computation is regulated by the descriptor similarity, thus, the top-retrieved references dissimilar to the query do not contribute to uncertainty estimation.

Extensive experiments in Chapter 5 demonstrated how this Spatial-Uncertainty-Estimation (SUE) outperformed other uncertainty estimation methods.

SQ4: Could we design strategies to densify reference maps in VPR that lead to accurate VPR-only localization?

Answer: Yes, we can. Chapter 6 presents different interpolation and extrapolation strategies to regress VPR descriptors at novel views and densify the reference map. Subsequently, VPR methods trained with a distance-based loss are reported to achieve accurate localization in these densified reference maps. Chapter 6 also illustrates how map densification using this Continuous Place-descriptor Regression (CoPR) is in fact complementary to relative pose estimation, where errors that cannot be resolved by relative pose estimation could be solved by map-densification. Unlike novel view synthesis in other parallel research fields, such as Neural Radiance Fields (NeRFs) and Gaussian Splatting, CoPR does not assume access to reference images but instead only operates on the compressed and robust feature descriptors available in VPR.

Main Research Question

MQ: How to achieve generalizability, confidence, and accuracy using VPR for localization?

Answer: The key element in this thesis that helped achieve generalizability, confidence, and accuracy in VPR has been the overlooked test-time *reference map*. This reference map is usually available offline and also contains poses, which is the primary assumption made in this thesis. It can be used to design frameworks that help improve generalization, uncertainty-estimation and localization accuracy of VPR methods.

Chapter 4 demonstrated how this reference map could be combined with Kornia augmentations to devise a new finetuning strategy (RSF) leading to domain-aware VPR descriptors. The RSF finetuned VPR methods are reported to generalize better to new domains. The poses of the references are exploited in Chapter 5 to develop a new confidence estimation method in VPR, the Spatial-Uncertainty-Estimation (SUE), which outperforms other categories of uncertainty estimation methods. SUE is also shown to be complementary to the expensive geometric verification, while being $100-1000\times$ computationally cheaper.

Chapter 6 argued that if the reference map is dense and the query trajectory is close to the reference trajectory, VPR can be used for accurate localization without a secondary relative pose estimation module. Building upon this insight, Continuous-Place-Descriptor-Regression (CoPR) is developed to regress VPR descriptors at novel views to densify the reference map. This descriptor regression is performed using various interpolation and extrapolation strategies, by treating the descriptors in the reference map as anchors. VPR methods trained with distance-based loss benefit the most from CoPR-densified reference maps. A key insight in Chapter 6 is that CoPR is also complementary to relative pose estimation, since the latter depends purely on the quality of retrieval which can be improved by densifying the reference maps.

7.3 BROADER DISCUSSION

This section provides a broader discussion within the scope of this thesis, but not necessarily limited to the experiments designed and performed in the preceding chapters.

The disconnect of VPR benchmarking from real-world use-cases

Most papers throughout VPR's history report results using metrics and datasets that are not representative of the different VPR applications. This is also not just limited to experimental setup in papers but instead extends to benchmarks, e.g., the benchmark presented in Chapter 3 and other benchmarks in VPR and image retrieval [9, 228]. A primary application that is motivated for VPR is loop-closure in SLAM, however, no existing benchmark in VPR directly evaluates how different methods could benefit SLAM. It is hoped that an increased Precision or Recall would directly translate to improved loop-closure in SLAM, however, this has not been systematically evaluated. Perhaps because this is not a trivial evaluation and requires dedicated datasets and metrics. Designing a VPR-focused SLAM framework could also be challenging in the choices of the odometry, filtering, and pose-graph optimization.

There is a disconnect between SLAM methods and VPR research. But the two can mutually re-inforce each other: false-positives from VPR could be filtered using temporal consistency provided by SLAM, and error drift in SLAM can be reduced with true-positives from VPR. Despite the well-motivated application, datasets in VPR essentially represent dense repeated traversals of trajectories and assume a map-based formulation, where as datasets in SLAM contain sparse revisiting of the same place where a map is assumed unavailable. The choices of evaluation metrics are then reflective of these assumptions.

This disconnect is also there in other VPR applications. For example, Structure-from-Motion (SfM) research also uses VPR to retrieve candidate images looking at the same scene and build a 3D representation. However, the popular go-to library for SfM, Colmap, does not use the recent advances in VPR [247]. Recently, this has been improved during the Large-scale Cross Device Localization (CrocoDL) Workshop[1] at ICCV 2025, where different modules (VPR, local feature extraction, local feature matching) were made available as an international challenge for the task of 6-DoF localization. The author also participated in the challenge with his unpublished VPR method named *Dera* which at the time of writing this thesis was the best performing VPR method[2].

Such evaluations of well-defined tasks, like VPR, local feature extraction, and feature matching, as modules of a larger application, e.g., 3D reconstruction and SLAM make it possible to quantify the end-benefits of advances in smaller tasks. This trend is still quite nascent, and problems exist, for example, benchmarking a VPR method with the reconstruction pipeline in the CrocoDL benchmark may take several days for a single evaluation. More efforts are nevertheless needed to bridge the advancements in VPR directly to its real-world applications, and in the corresponding evaluation pipelines.

The lack of research interest in false-positive prediction for VPR

This thesis argued (Chapter 3) for the importance of predicting false-positives and false-negatives in VPR research. This was further explored in depth in Chapter 5 where the various methods to estimate uncertainty in VPR were categorized into four categories: Retrieval-

[1] https://www.codabench.org/competitions/9471/
[2] https://github.com/MubarizZaffar/Dera

based Uncertainty Estimation (RUE), Deep-learning-based Uncertainty Estimation (DUE), Spatial Uncertainty Estimation (SUE), and Geometric Verification (GV). It was established that among the computationally feasible (i.e., other than GV) methods, SUE was the best performing predictor of false-positives in densely collected datasets.

However, if we look at the baselines for SUE in Chapter 5, it is evident that there is little research in false-positive prediction for VPR, as compared to the many works for improving accuracy of VPR methods. This perhaps also relates to the previous discussion on the disconnect between VPR benchmarking and real-world use-cases. For a real-world use-case, such as SLAM, a false-positive can be catastophic. Nevertheless, since existing VPR metrics and trends do not directly punish false-positives, the corresponding research interest in false-positive prediction has remained limited. *Given an acceptable-level of precision, knowing when your VPR system is failing is equally or perhaps even more important than failing less often.*

The reliance of relative pose estimation on VPR

The problem of relative pose estimation between two or multiple images has gained significant research interest recently either as the main task or one of the main tasks with many recent seminal works on this topic, such as DUST3R [248], MAST3R [22], Hloc [31], Pow3R [164], VGGT [163], and MapAnything [165], etc. All of these methods aim to estimate a relative pose between an anchor (or multiple anchors) and a query image: an important task that has applications for accurate 6-DoF localization and 3D reconstruction.

Nevertheless, the underlying assumption in these works is that the query and the anchor image share some visual overlap of the scene. For their respective evaluations, it is assumed that this overlap is given as a ground-truth. In a real-world application, where ground-truth visual overlap is unknown, VPR will be used to retrieve suitable anchors for a given query image. A failure to retrieve the correct reference image cannot be resolved by a relative pose estimation module. This reliance on VPR is a strong motivation to undertake research in VPR and was briefly discussed in Chapter 6. The author nevertheless believes that it deserves more through studying and attention, especially with benchmarks, challenges and metrics that can directly motivate improving VPR along the lines of robustness, re-ranking, and uncertainty estimation.

The fairness of assuming access to the test-time reference map in VPR

The conclusions drawn in this thesis rely on the assumption that the test-time reference map in available offline and perhaps also contains poses (ref. Chapter 5) of the reference images. This assumption allows the author to perform domain-adaptation in Chapter 4, estimate image-matching uncertainty in Chapter 5, and densify the reference maps for accurate localization in Chapter 6. A reader may wonder on the fairness of this assumption. It may or may not be fair to assume access to the test-time reference database, which mainly depends on the application of interest.

The author for this comment on fairness recaps the definitions and applications of VPR reviewed in the work of Garg et al. [233]. VPR can be primarily classified into *online and offline VPR*. Online VPR does not assume access to the complete reference database before a test-time query is received, but instead the reference set is built online as a robot/agent explores a given environment. On the other hand offline VPR assumes that the test-time

reference database has been pre-collected and is fully available offline. Most works in VPR, including this thesis, have been focused on offline VPR. The primary application of online VPR is in loop-closure for SLAM, whereas almost all other applications of VPR, such as image retrieval, landmark detection, map-based navigation, Structure-from-Motion (SfM), image cataloging, coarse localization, fall within offline VPR.

The benchmark designed in Chapter 3 (VPR-Bench) does not evaluate online VPR but instead is focused on offline VPR. Evaluating online VPR for loop-closure in SLAM remains an open research gap, as discussed previously. Chapter 4 (RSF) is directly relevant for offline VPR, and although it has not been evaluated for the online VPR setting, RSF can still be used: the VPR model is finetuned online in intervals as the robot collects new data. Chapter 5 (SUE) has been focused on offline VPR and cannot be used directly for online VPR, however, once the collected reference database is large enough (presumably several kilometers squared in an outdoor setting), SUE may be used to estimate image matching uncertainty. Chapter 6 can be used for both online and offline VPR, where the available anchors are individually densified, irrespective of whether the reference anchors were already available or collected as the robot continues to explore its environment.

Conclusively, for the case of offline VPR and its corresponding applications, the author believes it is fair to assume access to the test-time reference map, as has been assumed in all the preceding chapters. For online VPR, the experimental setup in Chapter 3 needs to be re-designed. The conclusions drawn in Chapters 4 and 5 remain relevant for online VPR but require further investigation. Chapter 6 remains agnostic to online or offline VPR.

7.4 FUTURE DIRECTIONS

This section presents a number of ideas that could be investigated in the future. These ideas are further categorized into proposals grounded in the content of this thesis's chapters and directions based on the author's broader reflections on the field.

7.4.1 IDEAS RELATED DIRECTLY TO THIS THESIS

A number of ideas that could serve as follow-up works for the research presented in this thesis are now discussed.

Investigating other types of augmentations for RSF

In Chapter 4, viewpoint and appearance augmentations from the Kornia library have been used. These augmentations include perspective transformations as a proxy for viewpoint-invariance and image transformations (color change, grayscaling, etc) as a proxy for appearance-invariance. However, these augmentations are quite basic and do not necessarily represent the kind of query-reference changes that are observed in real-world datasets, e.g., weather change and temporal changes. With the rise of image-to-image translation using Vision-Language-Models (VLMs), e.g., InstructPix2Pix [231], it is possible to create more realistic viewpoint and appearance changes for finetuning VPR methods with RSF.

Learning-based Spatial-Uncertainty-Estimation

SUE, as described in Chapter 5 requires two variables as hyper-parameters which are tuned on the Pittsburgh dataset [80] and then used as constants at test time. These hyper-parameters

include K and λ. For the K-best retrieved references, the hyper-parameter λ controls the non-linear relative contribution to uncertainty estimation of a pose p_i for the nearest neighbor $f_{(i)} \in \mathcal{R}_{\text{nn}}$ given its distance $d_{(i)}$ in the feature space. These two hyper-parameters K and λ do not need to be constants at test-time but can instead be variables of the feature descriptors, i.e., $K(\mathcal{R}_{\text{nn}})$ and $\lambda(\mathcal{R}_{\text{nn}})$. A neural network could be trained to estimate the distribution for these variables which could lead to improved generalization and better uncertainty estimation.

Using multiple anchors for CoPR
The Continuous Place-descriptor Regression (CoPR) performed in Chapter 6 uses only one anchor descriptor to regress the descriptor at a novel viewpoint. However, a single anchor is by no means a requirement/limitation of CoPR, since multiple anchors are readily available in the VPR reference map. Using multiple anchors has the benefit that each anchor may contain complementary information about the scene, and a neural network (e.g., a graph neural network or transformer) could learn to combine this complementary information from multiple anchors for regressing the novel descriptor. The author did in fact briefly experiment with this, however, for the chosen datasets there was no evident benefit. This relates to the nature of the anchors (sequential instead of uniformly spread-out) and the training data available for the chosen datasets, but for a different experimental setup where the scenes represent a larger scale and diverse viewpoints, combining information from multiple anchors could benefit the novel viewpoint descriptor regression.

Novel viewpoint selection for CoPR
In Chapter 6 the selection of viewpoints to regress the novel VPR descriptors was either based on uniform sampling around an anchor (7-scenes dataset) or based on prior information (Station Escalator dataset). Such selection of novel viewpoints could lead to a task of regressing descriptor at irrelevant poses. A wall (occlusion) could exist between an anchor's location and the chosen novel viewpoint to regress the VPR descriptor: the anchor does not contain useful information about the novel viewpoint. Future works could investigate methods to smartly select relevant novel viewpoints for descriptor regression that are directly beneficial for VPR and its various applications.

7.4.2 IDEAS BASED ON BROADER REFLECTION

The author here reflects on the existing literature and the contributions of this thesis and proposes some recommendations for the VPR community.

Benchmarking VPR directly for applications
The existing benchmarks in VPR have primarily evaluated it for metrics such as Precision and Recall [8], Recall@N [9], and Pose Accuracy [120]. These metrics do not evaluate how different VPR methods can benefit SLAM: an important application for VPR. The availability of SLAM frameworks, such as PySLAM [3], allows evaluating VPR directly for the task of loop-closure in SLAM. It should be studied which datasets are useful in this case, since an evaluation of VPR for SLAM would be relevant when there are many loop-closures in the dataset, and especially if some of these loop-closure candidates are challenging.

[3] https://github.com/luigifreda/pyslam

Punishing VPR methods with poor uncertainty estimates
The most common evaluation metric (Recall@N) in the experimental setups of SOTA VPR methods has been statistics that rank methods based on the percentage of retrieved true-positives. A true-positive does not necessarily need to be the best matching retrieved reference, but any of the Top-N retrievals. This evaluation metric does not use the uncertainty estimates of a VPR method.

Ranking methods based on the quality of uncertainty estimates is important for some applications, such as loop-closure. A VPR method that can fully distinguish between true- and false-positives but can only retrieve 50% of the total true-positives may be more suitable than a method that can retrieve 70% of the total true-positives but cannot distinguish well between true- and false-positives. Chapter 3 argues about the use of Precision-Recall (PR) curves and Receiver-Operator-Characteristics (ROC) curves which can evaluate uncertainty estimates of VPR methods. However, these metrics did not get picked up by the research community for evaluating SOTA VPR methods [6, 91, 222]. Perhaps this relates to the bias of following footsteps in experimental evaluation, but it is also an artifact of the lack of datasets and easy-to-use evaluation pipelines for uncertainty-critical applications of VPR. More work in needed in designing easy-to-use evaluation pipelines that encourage the VPR and image retrieval community to evaluate and improve uncertainty estimation in VPR.

Moving away from Google-Street-View for training VPR methods
The existing largest dataset for training deep-learning-based VPR methods is the GSV-Cities dataset [35] created using Google-Street-View: a platform that does not provide a permissible license. This dataset has been used to train all of the current SOTA VPR methods, such as BoQ [6], SALAD [91], CricaVPR [7], and MegaLoc [249]. The images collected using Google Street View do not contain significant appearance changes between images of the same place, and always represent similar viewpoint changes between the queries and references. This lack of diversity in this platform is indeed problematic for VPR research, where robustness against significant viewpoint and appearance changes is needed. Mapillary [4] on the other hand is a platform that provides much more diversity in terms of camera types, appearance changes and viewpoints. Queries and references from Mapillary could contain pedestrian to car viewpoint change, day to night appearance change, and camera types such as fisheye and spherical. However, the GPS ground-truth in Mapillary is poor, and can lead to images of different places labelled as the same place. Future works should create Mapillary-based datasets for VPR that are geometrically-verified with datasets having accurate GPS ground-truth, e.g., the Zenseact Open Dataset [250] and the Nvidia Autonomous Vehicle dataset [5]. Training VPR methods using such diversely collected data is critical for self-driving in new domains. Moreover, large-scale 3D reconstruction and especially in-the-wild applications can benefit from VPR methods trained on large-scale diversely-collected data.

[4] www.mapillary.com
[5] https://huggingface.co/datasets/nvidia/PhysicalAI-Autonomous-Vehicles

7.4.3 FINAL WORDS

VPR has been heavily researched in the past decade, and recent methods built on top of foundation model backbones and trained on large-scale datasets demonstrate robust performance. These methods are at times able to correctly match images that are even challenging for humans. Such recent advances have enabled applications of VPR for structure-from-motion [251], image retrieval, landmark detection and retrieval [129], and SLAM [252]. This success and robustness of SOTA VPR methods, in fact, span out new interesting applications, e.g., astronaut photography [253] and crime detection [6]. The viewpoint- and appearance-invariance challenges that were originally motivated as research questions in VPR and have been investigated for decades now seem somewhat solved by the state-of-the-art.

However, new challenges have emerged that require continued research attention. Of these challenges, the most important next step, in the author's opinion, is evaluating and improving VPR strictly and explicitly within the context of its applications. It has been shown that combining SOTA VPR methods with geometric verification and strong local feature matching provides benefits for large-scale 3D reconstruction, but it is not yet solved: please see the leader-board for the ICCV 2025 CrocoDL workshop[7]. Once the community starts evaluating VPR directly for applications, its central role for visual-based localization will become evident: this will help evaluate VPR for the primary motivation outlined in the introduction chapter 1.1.2.

Perceptual-aliasing has been discussed as a fundamental challenge dating almost 10 years back to the popular VPR survey [5], but it still remains an unsolved problem. Approaches that can successfully handle perceptual aliasing or predict false-positives are limited and far from what's needed. The need for viewpoint variance vs invariance in VPR, the quality of uncertainty estimates, and the benefits of map densification are all understudied problems, and more importantly, problems that are best studied within the context and framework of downstream applications. Thus, the author re-iterates that the most important next step, in the author's opinion, is evaluating and improving VPR strictly and explicitly within the context of its applications.

In conclusion, *VPR is not yet fully solved.*

[6] https://geospy.ai/

[7] https://www.codabench.org/competitions/9471/

BIBLIOGRAPHY

REFERENCES

[1] Oscar de Groot, Alberto Bertipaglia, Hidde Boekema, Vishrut Jain, Marcell Kegl, Varun Kotian, Ted Lentsch, Yancong Lin, Chrysovalanto Messiou, Emma Schippers, et al. A vehicle system for navigating among vulnerable road users including remote operation. *IEEE Intelligent Vehicles Symposium*, 2025.

[2] Rachel Metz. Google maps is using giant virtual arrows to stop people from getting lost, February 2019. [Retrieved; Sept-23-2025].

[3] Katharina Buchholz. These are the countries most affected by the decline in working age populations, January 2021. [Retrieved; Sept-22-2025].

[4] Nathan Piasco, Désiré Sidibé, Cédric Demonceaux, and Valérie Gouet-Brunet. A survey on visual-based localization: On the benefit of heterogeneous data. *Pattern Recognition*, 74:90–109, 2018.

[5] Stephanie Lowry, Niko Sünderhauf, Paul Newman, John J Leonard, David Cox, Peter Corke, and Michael J Milford. Visual place recognition: A survey. *IEEE Transactions on Robotics*, 32(1):1–19, 2015.

[6] Amar Ali-Bey, Brahim Chaib-draa, and Philippe Giguère. Boq: A place is worth a bag of learnable queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17794–17803, 2024.

[7] Feng Lu, Xiangyuan Lan, Lijun Zhang, Dongmei Jiang, Yaowei Wang, and Chun Yuan. Cricavpr: Cross-image correlation-aware representation learning for visual place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16772–16782, 2024.

[8] Mubariz Zaffar, Sourav Garg, Michael Milford, Julian Kooij, David Flynn, Klaus McDonald-Maier, and Shoaib Ehsan. VPR-Bench: An open-source visual place recognition evaluation framework with quantifiable viewpoint and appearance change. *International Journal of Computer Vision*, 129(7):2136–2174, 2021.

[9] Gabriele Berton, Riccardo Mereu, Gabriele Trivigno, Carlo Masone, Gabriela Csurka, Torsten Sattler, and Barbara Caputo. Deep visual geo-localization benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5396–5407, 2022.

[10] Sirisha Sirisha. Google maps is using giant virtual arrows to stop people from getting lost, June 2023. [Retrieved; Sept-25-2025].

[11] Andrew Rogers. Google maps usage statistics, Sept 2025. [Retrieved; Sept-25-2025].

[12] Elliott D Kaplan and Christopher Hegarty. *Understanding GPS/GNSS: principles and applications*. Artech house, 2017.

[13] Nils Ole Tippenhauer, Christina Pöpper, Kasper Bonne Rasmussen, and Srdjan Capkun. On the requirements for successful gps spoofing attacks. In *Proceedings of the 18th ACM conference on Computer and communications security*, pages 75–86, 2011.

[14] Ouster lidar sensors. `https://ouster.com/products/hardware/os1-lidar-sensor`. Accessed: 2025-11-18.

[15] Marco Tranzatto, Mihir Dharmadhikari, Lukas Bernreiter, Marco Camurri, Shehryar Khattak, Frank Mascarich, Patrick Pfreundschuh, David Wisth, Samuel Zimmermann, Mihir Kulkarni, et al. Team cerberus wins the darpa subterranean challenge: Technical overview and lessons learned. *Field Robotics*, 4:349–312, 2024.

[16] Tixiao Shan, Brendan Englot, Drew Meyers, Wei Wang, Carlo Ratti, and Daniela Rus. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 5135–5142. IEEE, 2020.

[17] Peter Biber and Wolfgang Straßer. The normal distributions transform: A new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, volume 3, pages 2743–2748. IEEE, 2003.

[18] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pages 241–257. Springer, 2016.

[19] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938–2946, 2015.

[20] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using rgb and poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5044–5053, 2023.

[21] Zakaria Laskar, Iaroslav Melekhov, Surya Kalia, and Juho Kannala. Camera relocalization by computing pairwise relative poses using convolutional neural network. In *IEEE International Conference on Computer Vision Workshops*, pages 929–938, 2017.

[22] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024.

[23] Sijie Zhu, Taojiannan Yang, and Chen Chen. Vigor: Cross-view image geo-localization beyond one-to-one retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3640–3649, 2021.

[24] Zimin Xia, Yujiao Shi, Hongdong Li, and Julian FP Kooij. Adapting fine-grained cross-view localization to areas without fine ground truth. In *European Conference on Computer Vision*, pages 397–415. Springer, 2024.

[25] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, 2016.

[26] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011.

[27] Johannes L Schönberger. *Robust methods for accurate and efficient 3D modeling from unstructured imagery*. PhD thesis, ETH Zurich, 2018.

[28] Patrik Schmuck and Margarita Chli. Multi-uav collaborative monocular slam. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3863–3870. IEEE, 2017.

[29] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of CNN-based absolute camera pose regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3302–3312, 2019.

[30] Sovann En, Alexis Lechervy, and Frédéric Jurie. Rpnet: An end-to-end network for relative camera pose estimation. In *European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[31] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019.

[32] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7199–7209, 2018.

[33] Paul-Edouard Sarlin, Ajaykumar Unagar, Mans Larsson, Hugo Germain, Carl Toft, Viktor Larsson, Marc Pollefeys, Vincent Lepetit, Lars Hammarstrand, Fredrik Kahl, et al. Back to the feature: Learning robust camera localization from pixels to pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3247–3257, 2021.

[34] Zelong Zeng, Zheng Wang, Fan Yang, and Shin'ichi Satoh. Geo-localization via ground-to-satellite cross-view image retrieval. *IEEE Transactions on Multimedia*, 25:2176–2188, 2022.

[35] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 513:194–203, 2022.

[36] Mubariz Zaffar, Liangliang Nan, Sebastian Scherer, and Julian FP Kooij. The overlooked value of test-time reference sets in visual place recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7234–7243, 2025.

[37] Mubariz Zaffar, Liangliang Nan, and Julian FP Kooij. On the estimation of image-matching uncertainty in visual place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17743–17753, 2024.

[38] Mubariz Zaffar, Liangliang Nan, and Julian Francisco Pieter Kooij. CoPR: Toward accurate visual localization with continuous place-descriptor regression. *IEEE Transactions on Robotics*, 2023.

[39] David G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[40] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *ECCV*, pages 404–417. Springer, 2006.

[41] Stephen Se, David Lowe, and Jim Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21(8):735–758, 2002.

[42] Henrik Andreasson and Tom Duckett. Topological localization for mobile robots using omni-directional vision and local features. *IFAC Proceedings Volumes*, 37(8):36–41, 2004.

[43] Elena Stumm, Christopher Mei, and Simon Lacroix. Probabilistic place recognition with covisibility maps. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4158–4163. IEEE, 2013.

[44] Jana Košecká, Fayin Li, and Xialong Yang. Global localization and relative positioning based on scale-invariant keypoints. *Robotics and Autonomous Systems*, 52(1):27–38, 2005.

[45] Ana Cris Murillo, José Jesús Guerrero, and C Sagues. Surf features for efficient robot localization with omnidirectional images. In *Proceedings of IEEE ICRA*, pages 3901–3907, 2007.

[46] Mark Cummins and Paul Newman. Appearance-only slam at large scale with fab-map 2.0. *IJRR*, 30(9):1100–1123, 2011.

[47] Will Maddern, Michael Milford, and Gordon Wyeth. Cat-slam: probabilistic localisation and mapping using a continuous appearance-based trajectory. *IJRR*, 31(4):429–451, 2012.

[48] Motilal Agrawal, Kurt Konolige, and Morten Rufus Blas. Censure: Center surround extremas for realtime feature detection and matching. In *European Conference on Computer Vision*, pages 102–115. Springer, 2008.

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

[50] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *ECCV*, pages 430–443. Springer, 2006.

[51] Christopher Mei, Gabe Sibley, Mark Cummins, Paul Newman, and Ian Reid. A constant-time efficient stereo slam system. In *Proceedings of the British machine vision conference*, volume 1. BMVA Press, 2009.

[52] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *null*, page 1470. IEEE, 2003.

[53] Adrien Angeli, Stéphane Doncieux, Jean-Arcady Meyer, and David Filliat. Incremental vision-based topological slam. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1031–1036. Ieee, 2008.

[54] Kin Leong Ho and Paul Newman. Detecting loop closure with scene sequences. *International Journal of Computer Vision*, 74(3):261–286, 2007.

[55] Junqiu Wang, Hongbin Zha, and Roberto Cipolla. Combining interest points and edges for content-based image retrieval. In *IEEE International Conference on Image Processing 2005*, volume 3, pages III–1256. IEEE, 2005.

[56] David Filliat. A visual bag of words method for interactive qualitative localization and mapping. In *ICRA*, pages 3921–3926. IEEE, 2007.

[57] Relja Arandjelović and Andrew Zisserman. Dislocation: Scalable descriptor distinctiveness for location recognition. In *Asian Conference on Computer Vision*, pages 188–204. Springer, 2014.

[58] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision*, pages 467–483. Springer, 2016.

[59] Jerome Revaud, Cesar De Souza, Martin Humenberger, and Philippe Weinzaepfel. R2d2: Reliable and repeatable detector and descriptor. In *Advances in Neural Information Processing Systems*, pages 12405–12415, 2019.

[60] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018.

[61] Mihai Dusmanu et al. D2-net: A trainable CNN for joint description and detection of local features. In *CVPR*, pages 8092–8101, 2019.

[62] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3456–3465, 2017.

[63] Aude Oliva and Antonio Torralba. Building the gist of a scene: The role of global image features in recognition. *Progress in brain research*, 155:23–36, 2006.

[64] Ana C Murillo and Jana Kosecka. Experiments in place recognition using gist panoramas. In *IEEE International Conference on Computer Vision Workshops*, pages 2196–2203. IEEE, 2009.

[65] Gautam Singh and J Kosecka. Visual loop closing using gist descriptors in manhattan world. In *ICRA Omnidirectional Vision Workshop*, pages 4042–4047, 2010.

[66] Niko Sünderhauf and Peter Protzel. Brief-gist-closing the loop by simple means. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1234–1241. IEEE, 2011.

[67] Michael Calonder, Vincent Lepetit, Mustafa Ozuysal, Tomasz Trzcinski, Christoph Strecha, and Pascal Fua. Brief: Computing a local binary descriptor very fast. *IEEE T-PAMI*, 34(7):1281–1298, 2011.

[68] Hernán Badino, Daniel Huber, and Takeo Kanade. Real-time topometric localization. In *ICRA*, pages 1635–1642. IEEE, 2012.

[69] Michael J Milford and Gordon F Wyeth. Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights. In *International Conference on Robotics and Automation*, pages 1643–1649. IEEE, 2012.

[70] Edward Pepperell, Peter I Corke, and Michael J Milford. All-environment visual place recognition with smart. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 1612–1618. IEEE, 2014.

[71] Colin McManus, Ben Upcroft, and Paul Newmann. Scene signatures: Localised and point-less features for localisation. *Robotics, Science and Systems Conference*, 2014.

[72] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1808–1817, 2015.

[73] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8601–8610, 2018.

[74] Akihiko Torii, Hajime Taira, Josef Sivic, Marc Pollefeys, Masatoshi Okutomi, Tomas Pajdla, and Torsten Sattler. Are large-scale 3d models really necessary for accurate visual localization? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[75] Mubariz Zaffar, Shoaib Ehsan, Michael Milford, and Klaus McDonald-Maier. Cohog: A light-weight, compute-efficient, and training-free visual place recognition technique for changing environments. *IEEE Robotics and Automation Letters*, 5(2):1835–1842, 2020.

[76] Zetao Chen, Obadiah Lam, Adam Jacobson, and Michael Milford. Convolutional neural network-based place recognition. *preprint arXiv:1411.1509*, 2014.

[77] Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Robert Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

[78] Zetao Chen et al. Deep learning features at scale for visual place recognition. In *ICRA*, pages 3223–3230. IEEE, 2017.

[79] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[80] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5307, 2016.

[81] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, pages 3304–3311. IEEE Computer Society, 2010.

[82] Nate Merrill and Guoquan Huang. Lightweight unsupervised deep loop closure. *arXiv preprint arXiv:1805.07703, Robotics Science and Systems Conference*, 2018.

[83] Bingyi Cao, Andre Araujo, and Jack Sim. Unifying deep local and global features for image search. *arXiv*, pages arXiv–2001, 2020.

[84] Oriane Siméoni, Yannis Avrithis, and Ondrej Chum. Local features and visual words emerge in activations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11651–11660, 2019.

[85] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning CNN image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1655–1668, 2018.

[86] Marvin Chancán, Luis Hernandez-Nunez, Ajay Narendra, Andrew B Barron, and Michael Milford. A hybrid compact neural architecture for visual place recognition. *IEEE Robotics and Automation Letters*, 5(2):993–1000, 2020.

[87] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, pages 1–31, 2024.

[88] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[89] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[90] Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*, 9(2):1286–1293, 2023.

[91] Sergio Izquierdo and Javier Civera. Optimal transport aggregation for visual place recognition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 17658–17668, 2024.

[92] Feng Lu, Lijun Zhang, Xiangyuan Lan, Shuting Dong, Yaowei Wang, and Chun Yuan. Towards seamless adaptation of pre-trained models for visual place recognition. *arXiv preprint arXiv:2402.14505*, 2024.

[93] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. Particular object retrieval with integral max-pooling of CNN activations. *arXiv:1511.05879, ICLR*, 2016.

[94] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision*, 124(2):237–254, 2017.

[95] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. Neural codes for image retrieval. In *European conference on computer vision*, pages 584–599. Springer, 2014.

[96] Jerome Revaud, Jon Almazán, Rafael S Rezende, and Cesar Roberto de Souza. Learning with average precision: Training image retrieval with a listwise loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5107–5116, 2019.

[97] Zetao Chen, Fabiola Maffra, Inkyu Sa, and Margarita Chli. Only look once, mining distinctive landmarks from convnet for visual place recognition. In *Proceedings of*

the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)), pages 9–16. IEEE, 2017.

[98] Zetao Chen, Lingqiao Liu, Inkyu Sa, Zongyuan Ge, and Margarita Chli. Learning context flexible attention model for long-term visual place recognition. *IEEE Robotics and Automation Letters*, 3(4):4015–4022, 2018.

[99] Ahmad Khaliq, Shoaib Ehsan, Zetao Chen, Michael Milford, and Klaus McDonald-Maier. A holistic visual place recognition approach using lightweight CNNs for significant viewpoint and appearance changes. *IEEE Transactions on Robotics*, 2019.

[100] Tomas Jenicek and Ondrej Chum. No fear of the dark: Image retrieval under varying illumination conditions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9696–9704, 2019.

[101] Relja Arandjelović and Andrew Zisserman. Visual vocabulary with a semantic twist. In *Asian Conference on Computer Vision*, pages 178–195. Springer, 2014.

[102] Arsalan Mousavian, Jana Košecká, and Jyh-Ming Lien. Semantically guided location recognition for outdoors scenes. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4882–4889. IEEE, 2015.

[103] Erik Stenborg, Carl Toft, and Lars Hammarstrand. Long-term visual localization using semantically segmented images. In *2018 IEEE ICRA*, pages 6484–6490, 2018.

[104] Johannes L Schönberger, Marc Pollefeys, Andreas Geiger, and Torsten Sattler. Semantic visual localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6896–6906, 2018.

[105] Tayyab Naseer, Gabriel L Oliveira, Thomas Brox, and Wolfram Burgard. Semantics-aware visual localization under challenging perceptual conditions. In *2017 IEEE ICRA*, pages 2614–2620, 2017.

[106] Yi Hou, Hong Zhang, and Shilin Zhou. Evaluation of object proposals and convnet features for landmark-based visual place recognition. *Journal of Intelligent & Robotic Systems*, 92(3-4):505–520, 2018.

[107] Sourav Garg, Niko Sünderhauf, Feras Dayoub, Douglas Morrison, Akansel Cosgun, Gustavo Carneiro, Qi Wu, Tat-Jun Chin, Ian Reid, Stephen Gould, Peter Corke, and Michael Milford. Semantics for robotic mapping, perception and interaction: A survey. *Found. Trends Robot.*, 8(1–2):1–224, 2020.

[108] Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 883–890, 2013.

[109] Ioannis Kostavelis and Antonios Gasteratos. Semantic mapping for mobile robotics tasks: A survey. *RAS*, 66:86–103, 2015.

[110] Ananth Ranganathan. Detecting and labeling places using runtime change-point detection and place labeling classifiers, October 15 2013. US Patent 8,559,717.

[111] Yogesh Girdhar and Gregory Dudek. Online navigation summaries. In *2010 IEEE International Conference on Robotics and Automation*, pages 5035–5040. IEEE, 2010.

[112] Rohan Paul, Dan Feldman, Daniela Rus, and Paul Newman. Visual precis generation using coresets. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1304–1311. IEEE, 2014.

[113] Mahmut Demir and H Isil Bozma. Automated place detection based on coherent segments. In *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, pages 71–76. IEEE, 2018.

[114] Elin A Topp and Henrik I Christensen. Detecting structural ambiguities and transitions during a guided tour. In *2008 IEEE International Conference on Robotics and Automation*, pages 2564–2570. IEEE, 2008.

[115] Mubariz Zaffar, Shoaib Ehsan, Michael Milford, and Klaus D McDonald-Maier. Memorable maps: A framework for re-defining places in visual place recognition. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[116] Stephen Hausler, Tobias Fischer, and Michael Milford. Unsupervised complementary-aware multi-process fusion for visual place recognition. *arXiv preprint arXiv:2112.04701*, 2021.

[117] Frederik Warburg, Martin Jørgensen, Javier Civera, and Søren Hauberg. Bayesian Triplet Loss: Uncertainty quantification in image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12158–12168, 2021.

[118] Kaiwen Cai, Chris Xiaoxuan Lu, and Xiaowei Huang. STUN: Self-teaching uncertainty estimation for place recognition. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6614–6621. IEEE, 2022.

[119] Petr Gronat, Guillaume Obozinski, Josef Sivic, and Tomas Pajdla. Learning and calibrating per-location classifiers for visual place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 907–914, 2013.

[120] Noé Pion, Martin Humenberger, Gabriela Csurka, Yohann Cabon, and Torsten Sattler. Benchmarking image retrieval for visual localization. In *International Conference on 3D Vision (3DV)*, pages 483–494. IEEE, 2020.

[121] Frank Dellaert, Wolfram Burgard, Dieter Fox, and Sebastian Thrun. Using the condensation algorithm for robust, vision-based mobile robot localization. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 588–594. IEEE, 1999.

[122] Bernhard Zeisl, Torsten Sattler, and Marc Pollefeys. Camera pose voting for large-scale image-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2704–2712, 2015.

[123] Amir Roshan Zamir and Mubarak Shah. Accurate image localization based on google maps street view. In *Proceedings of the European Conference on Computer Vision*, pages 255–268. Springer, 2010.

[124] Torsten Sattler, Michal Havlena, Konrad Schindler, and Marc Pollefeys. Large-scale location recognition and the geometric burstiness problem. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 1582–1590, 2016.

[125] Arthur Moreau, Nathan Piasco, Dzmitry Tsishkou, Bogdan Stanciulescu, and Arnaud de La Fortelle. Coordinet: uncertainty-aware pose regressor for reliable vehicle localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2229–2238, 2022.

[126] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4762–4769. IEEE, 2016.

[127] Valentin Peretroukhin, Brandon Wagstaff, Matthew Giamou, and Jonathan Kelly. Probabilistic regression of rotations using quaternion averaging and a deep multi-headed network. *arXiv preprint arXiv:1904.03182*, 2019.

[128] Frederik Warburg, Soren Hauberg, Manuel López-Antequera, Pau Gargallo, Yubin Kuang, and Javier Civera. Mapillary street-level sequences: A dataset for lifelong place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2626–2635, 2020.

[129] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2575–2584, 2020.

[130] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 573–580. IEEE, 2012.

[131] Henning Lategahn, Johannes Beck, Bernd Kitt, and Christoph Stiller. How to learn an illumination robust image feature for place recognition. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 285–291. IEEE, 2013.

[132] Titus Cieslewski and Davide Scaramuzza. Efficient decentralized visual place recognition from full-image descriptors. In *2017 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, pages 78–82. IEEE, 2017.

[133] Yawei Ye, Titus Cieslewski, Antonio Loquercio, and Davide Scaramuzza. Place recognition in semi-dense maps: Geometric and learning-based approaches. In *British Machine Vision Conference (BMVC)*, 2017.

[134] Luis G Camara and Libor Přeučil. Spatio-semantic convnet-based visual place recognition. In *2019 European Conference on Mobile Robots (ECMR)*, pages 1–8. IEEE, 2019.

[135] Mihnea-Alexandru Tomită, Mubariz Zaffar, Michael Milford, Klaus McDonald-Maier, and Shoaib Ehsan. Sequence-based filtering for visual route-based navigation: Analysing the benefits, trade-offs and design choices. *arXiv preprint arXiv:2103.01994*, 2021.

[136] Dmytro Mishkin, Michal Perdoch, and Jiri Matas. Place recognition with wxbs retrieval. In *CVPR 2015 workshop on visual place recognition in changing environments*, volume 30, 2015.

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

[138] Ben Talbot, Sourav Garg, and Michael Milford. Openseqslam2. 0: an open source toolbox for visual place recognition under changing conditions. In *Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 7758–7765. IEEE, 2018.

[139] Sourav Garg, Niko Suenderhauf, and Michael Milford. Lost? appearance-invariant place recognition for opposite viewpoints using visual semantics. In *Proceedings of Robotics: Science and Systems XIV*, 2018.

[140] Stephen Hausler, Adam Jacobson, and Michael Milford. Multi-process fusion: Visual place recognition using multiple image processing methods. *IEEE Robotics and Automation Letters*, 4(2):1924–1931, 2019.

[141] Horia Porav, Will Maddern, and Paul Newman. Adversarial training for adverse conditions: Robust metric localisation using appearance transfer. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1011–1018. IEEE, 2018.

[142] Bruno Ferrarini, Maria Waheed, Sania Waheed, Shoaib Ehsan, Michael J Milford, and Klaus D McDonald-Maier. Exploring performance bounds of visual place recognition using extended precision. *IEEE Robotics and Automation Letters*, 5(2):1688–1695, 2020.

[143] Florent Perronnin, Yan Liu, Jorge Sánchez, and Hervé Poirier. Large-scale image retrieval with compressed fisher vectors. In *2010 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3384–3391. IEEE, 2010.

[144] Mikaela Angelina Uy and Gim Hee Lee. PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4470–4479, 2018.

[145] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *European conference on computer vision*, pages 304–317. Springer, 2008.

[146] Gian Diego Tipaldi, Luciano Spinello, and Wolfram Burgard. Geometrical flirt phrases for large scale place recognition in 2d range data. In *2013 IEEE International Conference on Robotics and Automation*, pages 2693–2698. IEEE, 2013.

[147] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Image-based localization using hourglass networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 879–886, 2017.

[148] Ronald Clark, Sen Wang, Andrew Markham, Niki Trigoni, and Hongkai Wen. Vidloc: A deep spatio-temporal model for 6-dof video-clip relocalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6856–6864, 2017.

[149] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2616–2625, 2018.

[150] Eric Brachmann, Frank Michel, Alexander Krull, Michael Ying Yang, Stefan Gumhold, et al. Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3364–3372, 2016.

[151] Tommaso Cavallari, Stuart Golodetz, Nicholas A Lord, Julien Valentin, Luigi Di Stefano, and Philip HS Torr. On-the-fly adaptation of regression forests for online camera relocalisation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4457–4466, 2017.

[152] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11983–11992, 2020.

[153] Shuai Chen, Tommaso Cavallari, Victor Adrian Prisacariu, and Eric Brachmann. Map-relative pose regression for visual re-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20665–20674, 2024.

[154] Mingyu Ding, Zhe Wang, Jiankai Sun, Jianping Shi, and Ping Luo. Camnet: Coarse-to-fine retrieval for camera re-localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2871–2880, 2019.

[155] Dominik Winkelbauer, Maximilian Denninger, and Rudolph Triebel. Learning to localize in new environments from synthetic training data. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5840–5846. IEEE, 2021.

[156] Qunjie Zhou, Torsten Sattler, Marc Pollefeys, and Laura Leal-Taixe. To learn or not to learn: Visual localization from essential matrices. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3319–3326. IEEE, 2020.

[157] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16739–16752, 2025.

[158] Fadi Khatib, Yuval Margalit, Meirav Galun, and Ronen Basri. Leveraging image matching toward end-to-end relative camera pose regression. In *DAGM German Conference on Pattern Recognition*, pages 185–201. Springer, 2024.

[159] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021.

[160] Janine Thoma, Danda Pani Paudel, Ajad Chhatkuli, and Luc Van Gool. Geometrically mappable image features. *IEEE Robotics and Automation Letters*, 5(2):2062–2069, 2020.

[161] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4938–4947, 2020.

[162] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 17627–17638, 2023.

[163] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.

[164] Wonbong Jang, Philippe Weinzaepfel, Vincent Leroy, Lourdes Agapito, and Jerome Revaud. Pow3r: Empowering unconstrained 3d reconstruction with camera and scene priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1071–1081, 2025.

[165] Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction. *arXiv preprint arXiv:2509.13414*, 2025.

[166] Khang Truong Giang, Soohwan Song, and Sungho Jo. Learning to produce semi-dense correspondences for visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19468–19478, 2024.

[167] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In *European conference on computer vision*, pages 752–765. Springer, 2012.

[168] Torsten Sattler, Michal Havlena, Filip Radenovic, Konrad Schindler, and Marc Pollefeys. Hyperpoints and fine vocabularies for large-scale location recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2102–2110, 2015.

[169] Mayank Bansal, Harpreet S Sawhney, Hui Cheng, and Kostas Daniilidis. Geo-localization of street views with aerial image databases. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1125–1128, 2011.

[170] Tsung-Yi Lin, Yin Cui, Serge Belongie, and James Hays. Learning deep representations for ground-to-aerial geolocalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5007–5015, 2015.

[171] Scott Workman and Nathan Jacobs. On the location dependence of convolutional neural network features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 70–78, 2015.

[172] Sixing Hu, Mengdan Feng, Rang MH Nguyen, and Gim Hee Lee. Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7258–7267, 2018.

[173] Yujiao Shi, Liu Liu, Xin Yu, and Hongdong Li. Spatial-aware feature aggregation for image based cross-view geo-localization. *Advances in Neural Information Processing Systems*, 32, 2019.

[174] Krishna Regmi and Ali Borji. Cross-view image synthesis using conditional gans. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3501–3510, 2018.

[175] Hongji Yang, Xiufan Lu, and Yingying Zhu. Cross-view geo-localization with layer-to-layer transformer. *Advances in Neural Information Processing Systems*, 34:29009–29020, 2021.

[176] Sijie Zhu, Mubarak Shah, and Chen Chen. Transgeo: Transformer is all you need for cross-view image geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1162–1171, 2022.

[177] Yuxuan Hou, Yi Yang, Junbo Wang, and Mengyin Fu. Road extraction assisted offset regression method in cross-view image-based geo-localization. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2934–2940. IEEE, 2022.

[178] Florian Fervers, Sebastian Bullinger, Christoph Bodensteiner, Michael Arens, and Rainer Stiefelhagen. Uncertainty-aware vision-based metric cross-view geolocalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21621–21631, 2023.

[179] Paul-Edouard Sarlin, Eduard Trulls, Marc Pollefeys, Jan Hosang, and Simon Lynen. Snap: Self-supervised neural maps for visual positioning and semantic understanding. *Advances in Neural Information Processing Systems*, 36:7697–7729, 2023.

[180] Zimin Xia, Olaf Booij, and Julian FP Kooij. Convolutional cross-view pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3813–3831, 2023.

[181] Giorgos Tolias, Yannis Avrithis, and Hervé Jégou. Image search with selective match kernels: aggregation across single and multiple images. *International Journal of Computer Vision*, 116(3):247–261, 2016.

[182] Duncan P Robertson and Roberto Cipolla. An image-based system for urban navigation. In *Bmvc*, number 51, page 165. Citeseer, 2004.

[183] Edward Johns and Guang-Zhong Yang. From images to scenes: Compressing an image cluster into a single scene model for place recognition. In *2011 International Conference on Computer Vision*, pages 874–881. IEEE, 2011.

[184] Giorgos Tolias, Yannis Avrithis, and Hervé Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1401–1408, 2013.

[185] Friedrich Fraundorfer, Christopher Engels, and David Nistér. Topological mapping, localization and navigation using image collections. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3872–3877. IEEE, 2007.

[186] Anicetus Odo, Stephen McKenna, David Flynn, and Jan Vorstius. Towards the automatic visual monitoring of electricity pylons from aerial images. In *15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications 2020*, pages 566–573. SciTePress, 2020.

[187] Arren Glover. Day and night, left and right. *Zenodo*, March 2014.

[188] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[189] James Philbin, Ondrej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[190] Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, June 2018.

[191] Sindre Skrede. Nordland dataset. `https://bit.ly/2QVBOym`, 2013.

[192] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

[193] Henrik Aanæs, Anders Lindbjerg Dahl, and Kim Steenstrup Pedersen. Interesting interest points. *International Journal of Computer Vision*, 97(1):18–35, 2012.

[194] John Skinner, Sourav Garg, Niko Sünderhauf, Peter Corke, Ben Upcroft, and Michael Milford. High-fidelity simulation for evaluating robotic vision performance. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2737–2744. IEEE, 2016.

[195] Lukas Murmann, Michael Gharbi, Miika Aittala, and Fredo Durand. A multi-illumination dataset of indoor object appearance. In *2019 IEEE International Conference on Computer Vision (ICCV)*, Oct 2019.

[196] Carlo Masone and Barbara Caputo. A survey on deep visual place recognition. *IEEE Access*, 9:19516–19547, 2021.

[197] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, volume 1, pages 886–893. IEEE, 2005.

[198] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio 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*, pages 3234–3243, 2016.

[199] Mans Larsson, Erik Stenborg, Lars Hammarstrand, Marc Pollefeys, Torsten Sattler, and Fredrik Kahl. A cross-season correspondence dataset for robust semantic segmentation. In *CVPR*, pages 9532–9542, 2019.

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

[201] Raghavender Sahdev and John K Tsotsos. Indoor place recognition system for localization of mobile robots. In *2016 13th Conference on Computer and Robot Vision (CRV)*, pages 53–60. IEEE, 2016.

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

[203] James Mount and Michael Milford. 2d visual place recognition for domestic service robots at night. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4822–4829. IEEE, 2016.

[204] Titus Cieslewski, Siddharth Choudhary, and Davide Scaramuzza. Data-efficient decentralized visual slam. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2466–2473. IEEE, 2018.

[205] Mihnea-Alexandru Tomită, Mubariz Zaffar, Michael Milford, Klaus McDonald-Maier, and Shoaib Ehsan. Convsequential-slam: A sequence-based, training-less visual place recognition technique for changing environments. *arXiv preprint arXiv:2009.13454*, 2020.

[206] David M Chen, Georges Baatz, Kevin Köser, Sam S Tsai, Ramakrishna Vedantham, Timo Pylvänäinen, Kimmo Roimela, Xin Chen, Jeff Bach, Marc Pollefeys, et al. City-scale landmark identification on mobile devices. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 737–744. IEEE, 2011.

[207] Zetao Chen, Adam Jacobson, Uğur M Erdem, Michael E Hasselmo, and Michael Milford. Multi-scale bio-inspired place recognition. In *2014 IEEE international conference on robotics and automation (ICRA)*, pages 1895–1901. IEEE, 2014.

[208] Ankit Mohan, Reynold Bailey, Jonathan Waite, Jack Tumblin, Cindy Grimm, and Bobby Bodenheimer. Tabletop computed lighting for practical digital photography. *IEEE Transactions on Visualization and Computer Graphics*, 13(4):652–662, 2007.

[209] Lukas Murmann, Abe Davis, Jan Kautz, and Frédo Durand. Computational bounce flash for indoor portraits. *ACM Transactions on Graphics (TOG)*, 35(6):1–9, 2016.

[210] Mubariz Zaffar, Ahmad Khaliq, Shoaib Ehsan, Michael Milford, and Klaus McDonald-Maier. Levelling the playing field: A comprehensive comparison of visual place recognition approaches under changing conditions. *arXiv preprint arXiv:1903.09107, IEEE ICRA Workshop on Database Generation and Benchmarking*, 2019.

[211] Mubariz Zaffar, Ahmad Khaliq, Shoaib Ehsan, Michael Milford, Kostas Alexis, and Klaus McDonald-Maier. Are state-of-the-art visual place recognition techniques any good for aerial robotics? *arXiv preprint arXiv:1904.07967 ICRA 2019 Workshop on Aerial Robotics*, 2019.

[212] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics (TOG)*, 36(6):1–14, 2017.

[213] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-François Lalonde. Deep outdoor illumination estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7312–7321, 2017.

[214] Fan Zeng, Adam Jacobson, David Smith, Nigel Boswell, Thierry Peynot, and Michael Milford. Lookup: Vision-only real-time precise underground localisation for autonomous mining vehicles. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 1444–1450. IEEE, 2019.

[215] Janine Thoma, Danda Pani Paudel, Ajad Chhatkuli, Thomas Probst, and Luc Van Gool. Mapping, localization and path planning for image-based navigation using visual features and map. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7383–7391, 2019.

[216] Jianliang Zhu, Yunfeng Ai, Bin Tian, Dongpu Cao, and Sebastian Scherer. Visual place recognition in long-term and large-scale environment based on CNN feature. In *IEEE Intelligent Vehicles Symposium (IV)*, pages 1679–1685. IEEE, 2018.

[217] Gabriele Moreno Berton, Valerio Paolicelli, Carlo Masone, and Barbara Caputo. Adaptive-attentive geolocalization from few queries: A hybrid approach. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2918–2927, 2021.

[218] Muhammad Awais, Muzammal Naseer, Salman Khan, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Foundation models defining a new era in vision: a survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[219] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[220] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4893–4902, 2019.

[221] Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geolocalization for large-scale applications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4878–4888, 2022.

[222] Gabriele Berton, Gabriele Trivigno, Barbara Caputo, and Carlo Masone. Eigenplaces: Training viewpoint robust models for visual place recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11080–11090, 2023.

[223] Gabriele Berton, Carlo Masone, Valerio Paolicelli, and Barbara Caputo. Viewpoint invariant dense matching for visual geolocalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12169–12178, 2021.

[224] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Mixvpr: Feature mixing for visual place recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2998–3007, 2023.

[225] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Relative camera pose estimation using convolutional neural networks. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 675–687. Springer, 2017.

[226] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. Learning fine-grained image similarity with deep ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1386–1393, 2014.

[227] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5022–5030, 2019.

[228] Burak Yildiz, Seyran Khademi, Ronald Maria Siebes, and Jan Van Gemert. Amstertime: A visual place recognition benchmark dataset for severe domain shift. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 2749–2755. IEEE, 2022.

[229] Mark Cummins. Highly scalable appearance-only slam-fab-map 2.0. *In Proceedings of the Robotics: Sciences and Systems (RSS) Conference*, 2009.

[230] Edgar Riba, Dmytro Mishkin, Daniel Ponsa, Ethan Rublee, and Gary Bradski. Kornia: an open source differentiable computer vision library for pytorch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3674–3683, 2020.

[231] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[232] Xiaofeng Liu, Chaehwa Yoo, Fangxu Xing, Hyejin Oh, Georges El Fakhri, Je-Won Kang, Jonghye Woo, et al. Deep unsupervised domain adaptation: A review of recent advances and perspectives. *APSIPA Transactions on Signal and Information Processing*, 11(1), 2022.

[233] Sourav Garg, Tobias Fischer, and Michael Milford. Where is your place, visual place recognition? In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 2021.

[234] Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems*, 30, 2017.

[235] Michael J Milford and Gordon F Wyeth. Mapping a suburb with a single camera using a biologically inspired slam system. *IEEE Transactions on Robotics*, 24(5):1038–1053, 2008.

[236] Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. *Advances in Neural Information Processing Systems*, 31, 2018.

[237] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *Proceedings of the International Conference on Learning Representations*, 2016.

[238] Jan Knopp, Josef Sivic, and Tomas Pajdla. Avoiding confusing features in place recognition. In *Proceedings of the European Conference on Computer Vision*, pages 748–761. Springer, 2010.

[239] Carl Toft, Will Maddern, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, et al. Long-term visual localization revisited. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[240] Vassileios Balntas, Shuda Li, and Victor Prisacariu. Relocnet: Continuous metric learning relocalisation using neural nets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 751–767, 2018.

[241] Arthur Moreau, Nathan Piasco, Dzmitry Tsishkou, Bogdan Stanciulescu, and Arnaud de La Fortelle. Lens: Localization enhanced by nerf synthesis. In *Conference on Robot Learning*, pages 1347–1356. PMLR, 2022.

[242] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 173–179. IEEE, 2013.

[243] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[244] Eric Brachmann and Carsten Rother. Expert sample consensus applied to camera re-localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7525–7534, 2019.

[245] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136. IEEE, 2011.

[246] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[247] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[248] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024.

[249] Gabriele Berton and Carlo Masone. Megaloc: One retrieval to place them all. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2861–2867, 2025.

[250] Mina Alibeigi, William Ljungbergh, Adam Tonderski, Georg Hess, Adam Lilja, Carl Lindström, Daria Motorniuk, Junsheng Fu, Jenny Widahl, and Christoffer Petersson. Zenseact open dataset: A large-scale and diverse multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20178–20188, 2023.

[251] Paul-Edouard Sarlin, Mihai Dusmanu, Johannes L Schönberger, Pablo Speciale, Lukas Gruber, Viktor Larsson, Ondrej Miksik, and Marc Pollefeys. Lamar: Benchmarking localization and mapping for augmented reality. In *European Conference on Computer Vision*, pages 686–704. Springer, 2022.

[252] Luigi Freda. pyslam: An open-source, modular, and extensible framework for slam. *arXiv preprint arXiv:2502.11955*, 2025.

[253] Alex Stoken and Kenton Fisher. Find my astronaut photo: Automated localization and georectification of astronaut photography. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6196–6205, 2023.

GLOSSARY

APE Absolute Pose Estimation.

CoPR Continuous-Place-Descriptor-Regression.

CVL Cross-View Localization.

DoF Degrees-of-Freedom.

FPS Frame Per Second.

GNSS Global Navigation Satellite System.

GPS Global Positioning System.

GSV Google-Street-View.

HD High-Definition.

IMU Inertial Measurement Unit.

IVG Intelligent Vehicles Group.

LiDAR Light Detection and Ranging.

RADAR Radio Detection and Ranging.

RPE Relative Pose Estimation.

RSF Reference-set-Finetuning.

SBL Structure-based Localization.

SLAM Simultaneous Localization and Mapping.

SUE Spatial-Uncertainty-Estimation.

VBL Visual-based Localization.

VPR Visual Place Recognition.

Curriculum Vitæ

Mubariz Zaffar

Post-doctoral Researcher, MAVLab, The Netherlands

Highly motivated Ph.D. researcher with seven years of hands-on experience in deep learning, computer vision, and place recognition. Expert in representation learning, information retrieval, and large-scale deep-learning-based pose estimation. Proven track record in employing deep-learning for loop closure, tracking and registration, automated large-scale data labeling, uncertainty estimation, leadership and supervision, and interdisciplinary collaboration. Plays competitive badminton and volunteers for community service.

Professional Experience

Feb 2026 – Present
Post-doctoral Researcher, *MAVLab*, The Netherlands
- Working on autonomous drone landing at night time on moving ships using light-based fiducial markers.

Feb 2021 – Feb 2026
Ph.D. Candidate, *Delft University of Technology*, The Netherlands
- Published research in visual place recognition tackling domain adaptation, uncertainty estimation, accurate localization, and benchmarking. Supervised five successful master's theses and several bachelor's projects in the domain of deep learning for computer vision, vision-language models, and representation learning.

Apr 2024 – Jul 2024
Visiting Ph.D. Researcher, *Carnegie Mellon University*, USA
- Designed a domain adaptation pipeline for deep-learning-based visual place recognition with vision foundation models that is the current state-of-the-art on multiple public benchmarks.

Jun 2018 – Feb 2021
Research Officer, *University of Essex*, United Kingdom
- Published research in improving and benchmarking of visual place recognition methods. Conducted research in the estimation of the memorability of places.

Aug 2021 – Aug 2023 **Founder and President**, *Pakistan Youth and Student Association*, The Netherlands

- Founded Pakistan Youth and Student Association (PYSA) as a volunteer in my part-time at TU Delft to connect the student, employed, and business communities of Pakistanis in the Netherlands. Wrote the association charter, led the fund-raising efforts for victims of the 2022 Pakistan floods, managed a large number of community gatherings, organized a number of webinars for study and work aspirants in the Netherlands. Awarded the TU Delft Diversity and Inclusion Grant.

Jun 2016 – May 2018 **Research and Development Engineer**, *Skyelectric*, Pakistan

- Developed hardware and software for the battery management unit of their Smart Energy System. Grew towards a team-lead position in the R&D department.

Education

Feb 2021 – Feb 2026 **Ph.D. Mechanical Engineering**, *Delft University of Technology*, Delft, The Netherlands.

- Exploiting the test-time reference map for visual place recognition.

Oct 2017 – Mar 2020 **M.Sc. Computer Science and Electronics**, *University of Essex*, Colchester, United Kingdom.

- Visual place recognition using deep learning for autonomous robots.

Sep 2014 – Sep 2017 **B.Sc. Electrical Engineering**, *National University of Sciences and Technology*, Islamabad, Pakistan. (Rector's high achiever award)

- A machine-learning-based sensory device and mobile application for early diagnosis of diabetic peripheral neuropathy.

Technical Skills

Programming	Python *(Advanced)*, PyTorch *(Advanced)*, C++ *(Intermediate)*
Simulation, Tools	Linux, ROS, Git, Scikit-Learn, OpenCV, PCL, Latex, Docker
Languages	English *(Fluent)*, Urdu *(Native)*, Dutch *(Basic)*
Competencies	Deep-learning, localization, image classification and segmentation, prediction, 3D data processing, representation learning, research publication, experimental validation, leadership, team management, collaboration, technical writing, teaching & mentoring students, presenting research at international conferences to small and large audiences.

Collaborative & Leadership Skills

Leadership	Founded and led the Pakistan Youth and Student Association. Designed a charter and grew the association membership up to 150 active members. Collaborated with various stakeholders, including the Pakistan Business Forum Holland and the Embassy of Pakistan in the Netherlands.
Supervision	Actively mentored and led the research projects of six B.Sc. and five M.Sc. students, contributing to impactful academic outcomes.
Communication	Presented research at three top international conferences. Delivered two guest talks: a) on the invitation of Dr. Eduard Trulls at localization and mapping team, Google, Switzerland, b) on the invitation of Dr. Sebastian Scherer at CMU Robotics Institute, Pittsburgh, USA. Gave several lectures in the Robot Software Practicals course at TU Delft.
Collaboration	Co-organized a workshop on Simultaneous Localization and Mapping at the reputed RSS 2025 conference, Los Angeles, USA. Partnered with cross-functional teams to design the localization and mapping module for the TU Delft self-driving car.

Awards and Highlights

- Highlight poster (top 11.9%) at IEEE CVPR, Seattle, USA, 2024.
- Delft University Fund travel grant for research exchange in North America, Delft, The Netherlands, 2023.
- IEEE Robotics and Automation Letters (RA-L) outstanding reviewer award, Philadelphia, USA, 2021.
- Skyelectric high performer of the year, Islamabad, Pakistan, 2018.
- Rector's high achiever award, Islamabad, Pakistan, 2016.

List of Publications

On 16 March 2026, according to Google Scholar, my citation count is 609, and my h-index is 7. Below is a selected list of my publications.

1. ***Mubariz Zaffar***, Sebastian Scherer, Liangliang Nan and Julian F. P. Kooij: The Overlooked Value of Test-time Reference Set in Visual Place Recognition. Appeared in the proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, pp. 7234-7243. 2025.

2. Oscar de Groot,...,***Mubariz Zaffar***, ..., Dariu Gavrila: A Vehicle System for Navigating Among Vulnerable Road Users Including Remote Operation. Appeared in the proceedings of the IEEE Intelligent Vehicles Symposium (IVS), pp. 2482-2489. 2025.

3. ***Mubariz Zaffar***, Liangliang Nan and Julian F. P. Kooij: On the estimation of image-matching uncertainty in visual place recognition. Appeared in the proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 17743-17753. 2024.

4. ***Mubariz Zaffar***, Liangliang Nan and Julian F. P. Kooij: CoPR: Toward accurate visual localization with continuous place-descriptor regression. Published in the IEEE Transactions on Robotics (T-RO) 39, no. 4: 2825-2841. 2023.

5. ***Mubariz Zaffar***, Sourav Garg, Michael Milford, Julian F. P. Kooij, David Flynn, Klaus McDonald-Maier and Shoaib Ehsan: VPR-bench: An open-source visual place recognition evaluation framework with quantifiable viewpoint and appearance change. Published in the International Journal of Computer Vision (IJCV) 129, no. 7: 2136-2174. 2021.

6. ***Mubariz Zaffar***, Shoaib Ehsan, Michael Milford and Klaus McDonald-Maier: CoHog: A light-weight, compute-efficient, and training-free visual place recognition technique for changing environments. Published in the IEEE Robotics and Automation Letters (RA-L) 5, no. 2: 1835-1842. 2020.

7. ***Mubariz Zaffar***, Shoaib Ehsan, Michael Milford, and Klaus McDonald-Maier: Memorable Maps: A framework for re-defining places in visual place recognition. Published in the IEEE Transactions on Intelligent Transportation Systems (T-ITS) 22, no. 12: 7355-7369. 2020.

Included in this thesis.

ACKNOWLEDGMENTS

I find writing these acknowledgments more difficult (but immensely enjoyable) than writing either the thesis introduction or the conclusions; for there are so many people that I want to acknowledge, so much I wish to write for them, and so much to be thankful for; but I have a limited space. I must not forget anyone and I have thus tried my best that I do not.

I would like to kick-off by acknowledging my brilliant supervisor Dr. Julian F. P. Kooij. Julian, I was lucky to have you as my Ph.D. supervisor. I say this with much weightage and from a both a personal and a professional perspective. I learned a lot from you, you are an amazing teacher with great patience, much knowledge and exceptional analytical capabilities. I always found myself chasing your analytical pace and your knowledge base, and even after these five years I believe I am still chasing it. Your academic success is publicly quite evident for everyone, but what some people might not know too well is how amazing you are as a person and given your value-system. You were there for me at the most difficult time of my life, shielding me from all the academic stress and even taking over menial tasks from me, so I could manage my personal struggles. If you had not been there, and if you had not been the person you are, I am not sure how I would have survived through that time, and if I would have even made it to this point today. I will forever be grateful to you, will continue to look up to you, and try to develop into a person that is a decent approximation of many of your values. I hope that you continue to be the same person in life, and the same advisor for the many Ph.D. candidates I expect you will supervise in the coming years.

I am thankful to my amazing co-supervisor Dr. Liangliang Nan, a person I found to be immensely kind, really hospitable, highly competent, and wise. Thank you for all your feedback on my research and my writings, which greatly helped to achieve the outcomes of this thesis. The BBQs at your home were great, I always enjoyed being there; you are a truly generous, welcoming and empathetic person. Many thanks for all your advice on life, I found it helpful and will continue to use it in the coming time. To our group head, Prof. Dr. Dariu Gavrila, thank you for considering me as a Ph.D. in your group, for funding my various work-related travels that allowed me to present my research at international forums, and for the latent support by acquiring various computational platforms for research and for the IV self-driving Prius. Congratulations to you on establishing one of Europe's finest academic research group in self-driving, and I hope that in the coming time you continue to take the steps that make it one of (if not) the best in the world.

To the other academics in our group, Dr. Barys Shyrokau, Dr. Georgios Papaioannou, and Prof. Dr. Riender Happee, thank you for all your advice on the various professional and personal matters, whenever we shared lunch sessions, birthday celebrations, and the various other occassions our group finds to have cakes. I wish you success in your academic life. To my favorite German, Dr. Holger Caesar, many thanks for your guidance on career and for all your help in the various steps related to it. Thank you for your advice on life, for your feedback on my research, and my professional outlook. I always found you helpful, approachable, and kind. I hope that we manage to collaborate on amazing research in the

coming time and that we end-up materializing the plans we are making these days. Thanks to the ME secretariaat, especially Hanneke, for all your support with administrative matters.

Much thanks to my colleagues in the first part of my Ph.D. journey: Zimin, Hidde, Xiaolin, Chrys, Vishrut, Varun, Alberto, Tugrul, and Andras. I had a truly great time with all of you, your presence made the group enjoyable, and the Ph.D. less stressful and difficult than it would have been otherwise. With each of you I share a connection: some inviting me to their homes, some teaching me the basics of computer vision and machine learning, some I could do almost any (dark) joke with, and some with whom I could vent. I wish all of you the best and I hope that you all continue to be successful in your professional and personal lives. Many thanks to the superman and batman of our group, Ronald and Mario, who helped me with various hands-on tasks related to my research, teaching, the IV compute platforms and the Prius. Our Ph.D. experiencs would have been immensely difficult if we did not have the two of you in our group. To my lovely colleagues from the 3DUU lab: Nail, Shenglan, and Shiming, thanks to all three of you for being there. Your presence in the past five years has been immensely pleasant, and I truly enjoyed the times we went out together. It was a pleasure to share the office with you: Jetze, Ted, Yancong, Thomas, Raj, Ceren and Jules. You were all great office mates. I found our office conversations about work and life truly interesting and inspiring, and I learned from each of you. Finally, to all the people in this paragraph: there is so much that I want to write for each of you but I am afraid it will end up being multiple pages. Trust me, I thought of each of you with much gratitude when I wrote your name here and realized that for everyone I could have easily written a unique personal note given our shared connection.

For the people in IV group who joined towards the end or with whom I had lower time overlap: Ferzam, Guopeng, Ali, Giacomo, Chunpeng, Joseph, Ziqi, Marko, Charlotte and Oscar, it was great to have been your colleague albeit it was a short time together. I wish you much success in your life and careers. I am thankful to Dr. Sebastian Scherer (a.k.a. Basti) for hosting me at the AIRLab, Caregie Mellon University, during the summer of 2024. I had a great time at your lab and I learned a lot while being there. To my supervisors in the past, Dr. Shoaib Ehsan, Prof. Dr. Klaus McDonald-Maier, Prof. Dr. Farrukh Kamran, and Dr. Hammad Mehmood Cheema, I am thankful for your guidance and support that carried me to undertake this Ph.D. journey. I do not expect to have received this Ph.D. offer from Julian, if I did not have you during my professional life. To my amazing external collaborators: Dr. Sourav Garg, Dr. Tobias Fischer, and Prof. Dr. Michael Milford, thank you for being there whenever I needed advice on research and on career. I learned from all three of you and I hope that we continue to collaborate. To my two amazing mentors and teachers while growing up: Shadab Fatima and Imran Khan, thanks to both of you for developing me from an early age. We started off quite far from here, but somehow you managed to carry me to this point; I will forever be grateful for the role you have played in my life.

Nevertheless, a Ph.D. journey does not just require a good professional support but also an immense support from the people in our daily lives. Thus, I must now shift focus to them and start by noting this down: I miss you ammi (mom). If there was someone in my life whom I could go to for any difficulty, personal or professional, and always find comfort, it was you. I lost you half way through my Ph.D.; I truly wish that you were standing there with me at my Ph.D. defense. Thank you for everything you did for the three of us; I wish that I am able to be the person you wanted me to be. My father, Zaffarullah Khan Saddozai,

whose entire ambition in his life has been to ensure high-quality education for his kids; it is your resilience and clear directions that carried the three of us forward in our academic life. I am thankful to you for supporting our education with everything you had and for the vision you had about our careers. My two sister, Dr. Tehreem Zaffar and Zahwa Zaffar, you make my personal life complete. Your support has been truly helpful all these years, and if it were not the two of you managing things back home, I would have had to perhaps stop my Ph.D. during those difficult years. I am grateful for all your unconditional support and love. And now to my significant other, my lovely wife, Dr. Sumbal Khan, you are one of the greatest things that happened to me in my life. Your support and love each day keeps me going and makes my life brighter. I am thankful to you for cheering me up, for taking care of me, for making my life happier and complete.

Thanks to my cousin Dr. Nosheen Latif and Dr. Hassan Khan, who took care of me while in Europe and who made me feel at home here. Thank you Jan bhai for looking after my family in Pakistan during those difficult years while encouraging me to continue my Ph.D. in the Netherlands. The support I received from the three of you in the various forms has had a huge impact on my Ph.D. trajectory, and I will forever be grateful. I am also thankful to my uncles Rashidullah Khan Saddozai and Latifullah Khan Saddozai for their role in my education and life. I am thankful to the board members, team members and volunteers of the Pakistani-Youth-and-Student-Association (PYSA-Delft), who made my role as a President successful. Running this volunteer organization at TU Delft also taught me many lessons and skills that proved helpful in my Ph.D.

During my time in Delft, I had a family away from home and it centered around two amazing people: Dr. Muhammad Mohsan and Dr. Aitazaz Ali Raja. If Julian shielded me in my professional life during those difficult years, I had the two of you supporting me in my personal life. I owe you both much, thanks for everything you have both done for me. I truly found two brothers in you. Much thanks to my close friends in the Pakistani community in Delft, including Aftab bhai, Samad, Tanveer bhai, Qasim bhai, Hussam bhai, Haider, Hassan, Osama bhai, Rehan, Usman bhai, Haris, Muneeb, Tayyaba and Sadia; your presence in/around Delft made my time here truly enjoyable and unforgettable. My dearest friends Abdul Wahab, Bilal, Fowad bhai, Hateem, Haris bhai, Minhaj, Asif, Khaqan bhai, Amir, Zain, Mussadiq bhai, Sami, Shahzeb, Sheryar, and Hassan, our get-togethers have been the place where I could relieve my Ph.D. stress and re-energize. Thanks to each of you. As you may note, there are many names in this paragraph, and I feel truly lucky that this is the case. It is these names that made a huge impact in my social life during the time in Delft.

Last but not the least, I am immensely thankful to the TU Delft AI Labs and Talent Programme which funded my Ph.D. and enabled this thesis. The Delft University Travel Fund and the TU Delft Transport and Mobility Institute funded several of my travels and stays during this Ph.D, I am thankful. I hope to have not forgotten anyone here. If I did, please know that it would only be in writing but I would know it in my heart. I end this section with some Urdu poetry I wrote during my research exchange at CMU, Pittsburgh, sitting next to the statue *Walking to the Sky*.

Mubariz
Delft, March 2026

Figure 7.1: Walking to the Sky statue by Jonathan Borofsky at Carnegie Mellon University (CMU), Pittsburgh, USA. Picture taken by the author during his research visit at CMU in the summer of 2024.

رکیں تو بھلا کیسے رکیں اب اس جہاں میں،
کہ پھسل کر آن پڑیں پھر ایک میدان میں۔

پوچھا کہ چلیں تو آخر کہاں چلیں،
کہتے، آؤ چلیں جہاں اس کے متقی انسان چلیں۔

— مبارز

How could we possibly stop in this world,
for we might slip and fall back into yet another field?

I asked, “If we must go, then where should we go?”
They said, “Come—let us go where the humbled go.”

— Mubariz