International Journal of Computer Vision (2021) 129:2136–2174
https://doi.org/10.1007/s11263-021-01469-5

VPR-Bench: An Open-Source Visual Place Recognition Evaluation Framework with Quantifiable Viewpoint and Appearance Change

Mubariz Zaffar[1,2] · Sourav Garg[3] · Michael Milford[3] · Julian Kooij[2] · David Flynn[4] · Klaus McDonald-Maier[1] · Shoaib Ehsan[1]

Received: 15 May 2020 / Accepted: 7 April 2021 / Published online: 7 May 2021

Abstract
Visual place recognition (VPR) is the process of recognising a previously visited place using visual information, often under varying appearance conditions and viewpoint changes and with computational constraints. VPR is related to the concepts of localisation, loop closure, 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. In this paper, we address these gaps through a new comprehensive open-source framework for assessing the performance of VPR techniques, dubbed "VPR-Bench". VPR-Bench (Open-sourced at: https://github.com/MubarizZaffar/VPR-Bench) 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. We apply and analyse popular evaluation metrics for VPR from both the computer vision and robotics communities, and discuss how these different metrics complement and/or replace each other, depending upon the underlying applications and system requirements. Our 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, we identify 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 (e) directional illumination change has more adverse effects on matching confidence than uniform illumination change. We also present 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.

Keywords Visual place recognition · SLAM · Autonomous robotics · Robotic vision

Communicated by Daniel Kondermann.

✉ Mubariz Zaffar
mubariz.zaffar@essex.ac.uk; m.zaffar@tudelft.nl

Sourav Garg
s.garg@qut.edu.au

Michael Milford
michael.milford@qut.edu.au

Julian Kooij
j.f.p.kooij@tudelft.nl

David Flynn
D.Flynn@hw.ac.uk

Klaus McDonald-Maier
kdm@essex.ac.uk

Shoaib Ehsan
sehsan@essex.ac.uk

[1] School of Computer Science and Electronic Engineering, University of Essex, Colchester CO4 3SQ, UK

1 Introduction

Visual place recognition (VPR) is a challenging and widely investigated problem within the computer vision community (Lowry et al. 2015). It identifies the ability of a system to match a previously visited place using on-board computer vision prowess, 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 (Cadena et al. 2016) to correct for localization drifts, image search based on visual content (Tolias et al. 2016a), location-refinement given human–machine interfaces (Robertson and Cipolla 2004), query-expansion (Johns and Yang 2011), improved representations (Tolias et al. 2013), vehicular navigation (Fraundorfer et al. 2007), asset-management using aerial imagery (Odo et al. 2020) and 3D-model creation (Agarwal et al. 2011).

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. 1).

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 (Glover 2014), ESSEX3IN1 (Zaffar et al. 2018) dataset, Campus Loop dataset (Merrill and Huang 2018) 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 (Torii et al. 2013), Oxford Buildings dataset (Philbin et al. 2007), Paris dataset (Philbin et al. 2008) and their revisited versions with increased 1M distractors by Radenović et al. (2018).[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 our work is attempting to reduce this gap by integrating datasets, metrics and techniques from both the domains into a novel framework 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. We utilise the detailed variation-quantified Point Feature dataset (Aanæs et al. 2012) and integrate it into our 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 Sect. 3.5. Since the Point Features dataset is a synthetically-created dataset, we also include the QUT multi-lane dataset (Skinner et al. 2016) and MIT multi-illumination dataset (Murmann et al. 2019), which each respectively represent quantified variations in viewpoint and illumination in a real-world setting.

[2] Cognitive Robotics, TU Delft, 2628CD Delft, Netherlands

[3] School of Electrical Engineering and Computer Science, Queensland University of Technology, Brisbane, QLD 4000, Australia

[4] School of Engineering and Physical Sciences, Smart Systems Group, Heriot-Watt University, Currie, Edinburgh EH14 4AS, United Kingdom

[1] These remarks are only depicting the evident trends and are not absolute. Large-scale datasets (e.g. the Nordland dataset by Skrede 2013 and Oxford robot-car dataset by Maddern et al. 2017) for the robotics community, and small-scale datasets (e.g. the INRIA Holidays dataset by Jegou et al. 2008) for the computer vision community do exist.

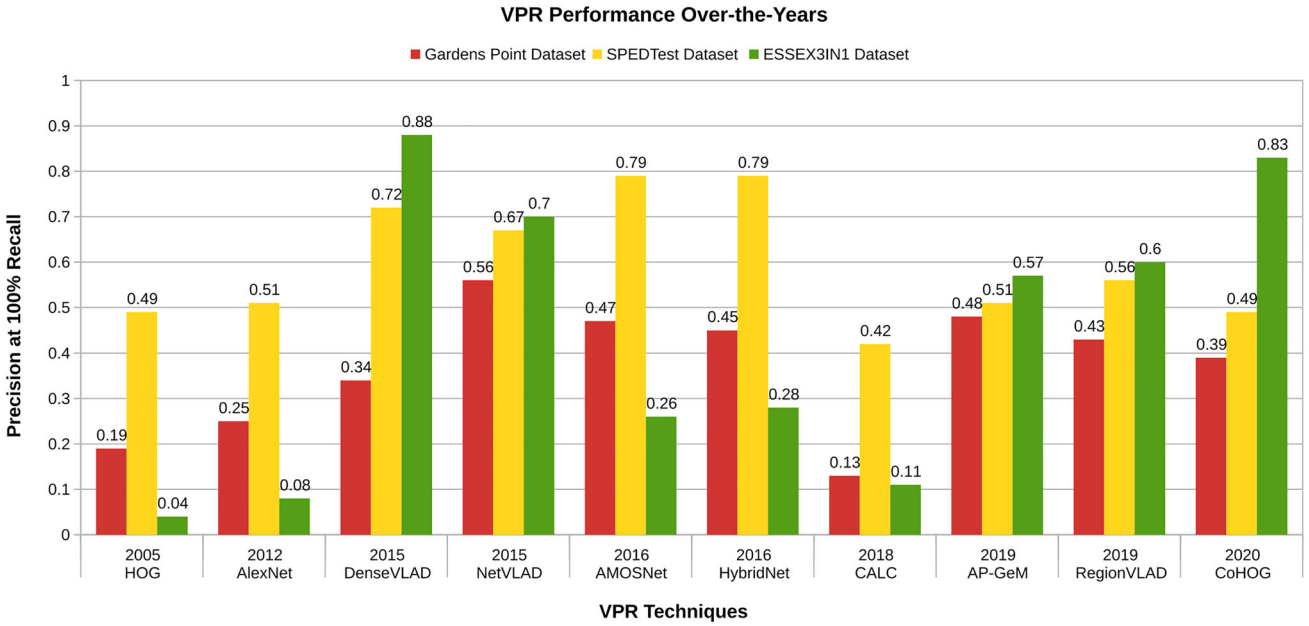

Fig. 1 Precision at 100% Recall (equivalent to RecallRate@1) of 10 VPR techniques on Gardens Point dataset (Chen et al. 2014b), SPEDTest dataset (Chen et al. 2018) and ESSEX3IN1 dataset (Zaffar et al. 2018) is shown here in a chronological order. The trends show irregularities in between techniques and datasets, while the increase in precision is also not temporally consistent. These datasets and techniques have been discussed later in our paper. 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

Furthermore, we take the opportunity to present a detailed meta-analysis enabled by VPR-Bench. We have integrated receiver-operating-characteristic (ROC) curves into VPR-Bench to analyse the ability of VPR techniques to find 'new places', i.e. true-negatives, which are generally not available in Precision-Recall type metrics. We perform experiments and present analysis on the distribution of true-positives within a sequence in our work, 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, we also discuss case-studies on ground-truth manipulation that can lead to varying state-of-the-art, and the CPU versus GPU performance differences for deep-learning-based VPR techniques. The descriptor size of VPR techniques also affects VPR performance and we analyse these effects in our work. 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 our comprehensive framework will be made fully public, and all datasets with their associated ground-truths will be re-released. An overview of our framework is shown in Fig. 2.

In summary, our main contributions are:

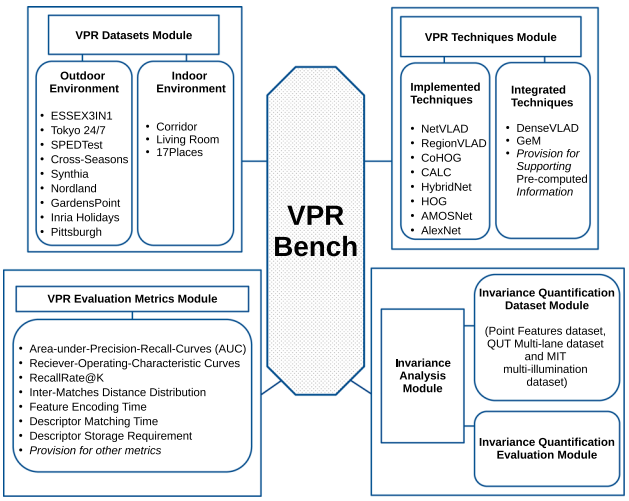

Fig. 2 A block-diagram overview of the developed VPR-Bench framework is shown here. All modules can be inter-linked within the framework and can also be independently modified for graceful updates in the future

1. We present 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 we 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. We present an open-source, fully-integrated, extensive framework for evaluating VPR performance. We re-implement a number of VPR techniques based on our unified templates and re-structure datasets and their ground-truths into consistent and compatible formats, which we will be re-releasing, 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. We quantify the notion of viewpoint and illumination invariance of VPR techniques by employing a detailed variation-quantified Point Features dataset. We then further extend our findings to 2 real-world, variation-quantified datasets, namely QUT multi-lane dataset and MIT multi-illumination dataset.
4. We present 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 versus invariance, the effects of descriptor size on the performance of a technique, the CPU versus GPU computational performance rankings and the trends of image retrieval times' variation with changing map-size on par with a platform's dynamics.

The remainder of the paper is organized as follows. In Sect. 2, a comprehensive literature review regarding VPR state-of-the-art is presented. Section 3 presents the details of the evaluation setup employed in this work. Section 4 puts forth the results and analysis obtained by evaluating the contemporary VPR techniques on public VPR datasets, along with insights into invariance quantification. Finally, conclusions and future directions are presented in Sect. 5.

2 Literature Review

The detailed theory behind visual place recognition (VPR), its challenges, applications, proposed techniques, datasets and evaluation metrics have been thoroughly reviewed by Lowry et al. (2015), and more recently by Garg et al. (2021), Zhang et al. (2021) and Masone and Caputo (2021).

Before diving deep into the core VPR literature review, it is important to co-relate and distinguish VPR research from closely related topics including visual-SLAM, visual-localisation and image matching (or correspondence problem), to set the scope of our research. A huge body of robotics research in the past few decades has been dedicated to the problem of simultaneously localising and mapping an environment, as thoroughly reviewed by Cadena et al. (2016). Performing SLAM with only visual information is called visual-SLAM, and Davison et al. (2007) were the first to fully demonstrate this. The localisation part of visual-SLAM can be broadly divided into two tasks: (1) Computing change in camera/robot pose while performing a particular motion, using inter-frame(s) co-observed information, (2) Recognising a previously seen place to perform loop-closure. The former is usually referred to as visual-localisation and Nardi et al. (2015) developed an open-source framework in this context for evaluating visual-SLAM algorithms. The latter is essentially an image-retrieval problem in the computer vision community, and within the context of robotics has been referred to as Visual Place Recognition (Lowry et al. 2015). Image matching (also referred to as keypoint matching or correspondence problem in some literature) consists of finding repeatable, distinct and static features in images, describing them using condition-invariant descriptors and then trying to locate co-observed features in various images of the same scene. It is primarily targeted for visual-localisation, 3D-model creation, Structure-from-Motion and geometric-verification, but can also be utilised for VPR. Jin et al. (2020) developed an evaluation framework along these lines for matching images across wide baselines. It is important to note here that image matching can also be included as a sub-module of a VPR system. Torii et al. (2019) demonstrated that such a system can achieve accurate localisation without the need for large-scale 3D-models.

VPR has therefore generally been approached as a retrieval problem that focuses on retrieving a correct match (either as the best-match or among the Top-N matches) from a reference database given a query image, under varying viewpoint and conditions. However, VPR may also be combined with local-feature matching (geometric verification) to perform highly accurate localisation at increased computational cost, as shown by Sattler et al. (2016), Camara et al. (2019) and Sarlin et al. (2019). 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, computational requirements and approach salience.

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 Lowe 2004) and Speeded Up Robust Features (SURF Bay et al. 2006). These descriptors have been used for the VPR problem by Se et al. (2002), Andreasson and Duckett (2004), Stumm et al. (2013), Košecká et al. (2005) and Murillo et al. (2007). A probabilistic visual-SLAM algorithm was presented by Cummins and Newman (2011)), 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 Maddern et al. (2012) to achieve Continuous Appearance Trajectory-based SLAM (CAT-SLAM) using a Rao–Blackwellised particle filter. CenSurE (Center Surround Extremas by Agrawal et al. (2008)) is another popular local feature descriptor and which has been used for VPR by Konolige and Agrawal (2008). FAST (Rosten and Drummond 2006) is a popular high speed corner detector that has been used in combination with the SIFT descriptor for SLAM by Mei et al. (2009). Matching of local feature descriptors is a computationally intense process which has been addressed by the Bag of visual Words (BoW Sivic and Zisserman 2003) 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 Angeli et al. (2008), Ho and Newman (2007), Wang et al. (2005) and Filliat (2007). Arandjelović and Zisserman (2014a) present 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 (Yi et al. 2016), R2D2 Revaud et al. (2019b), SuperPoint (DeTone et al. 2018) and D2-net (Dusmanu et al. 2019). Noh et al. (2017) designed a deep-learning-based local feature extractor and descriptor, namely DELF, that is used with geometric verification for large-scale image retrieval.

Global Feature Descriptors-based: VPR Global feature descriptors create a holistic signature for an entire image and Gist (Oliva and Torralba 2006) is one of the most popular global feature descriptor. Working on panoramic images, Murillo and Kosecka (2009), Singh and Kosecka (2010) used Gist for VPR. Sünderhauf and Protzel (2011) combined Gist with BRIEF (Calonder et al. 2011) to perform large scale visual-SLAM. Badino et al. (2012) 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 (Milford and Wyeth 2012) uses normalized pixel-intensity matching in a global fashion to perform VPR in challenging conditionally-variant environments. The original Seq-SLAM algorithm assumes constant speed of the robotic platform, thus, Pepperell et al. (2014) extended Seq-SLAM to consider variable speed instead. McManus et al. (2014) extract scene signatures from an image by utilising some *a priori* environment information and describe them using HOG-descriptors. DenseVLAD presented by Torii et al. (2015) 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 Sattler et al. (2018) and Torii et al. (2019). A more recent usage of traditional handcrafted feature descriptors for VPR was presented in CoHOG (Zaffar et al. 2020) which focuses on entropy-rich regions in an image and uses HOG as the regional descriptor for convolutional-regional matching.

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, Chen et al. (2014b) used features from the Overfeat Network (Sermanet et al. 2014) and combined it with the spatial filtering scheme of Seq-SLAM. This work was followed up by Chen et al. (2017b), 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 (Krizhevsky et al. 2012). An end-to-end neural-network-based holistic descriptor NetVLAD is introduced by Arandjelovic et al. (2016), where a new VLAD (Vector-of-Locally-Aggregated-Descriptors Jégou et al. 2010) layer is integrated into the CNN architecture achieving excellent place recognition results. A convolutional auto-encoder network is trained in an unsupervised fashion by Merrill and Huang (2018), utilizing HOG-descriptors of images and synthetic viewpoint variations for training. The work of Noh et al. (2017) was extended to DELG (DEep Local and Global Features by Cao et al. (2020)) combining generalized mean pooling for global descriptors and attention mechanism for local features. Recently, Siméoni et al. (2019) 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 Radenović et al. (2018) (GeM) introduces a new trainable 'Generalised Mean' layer into the deep image-retrieval architecture which has been shown to provide a performance boost. Chancán et al. (2020) 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.

Regions-of-Interest-focused: VPR Researchers have 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 Tolias et al. (2016b), where max-pooling across cropped areas in CNN layers' features define/extract ROIs. This work on R-MAC is further advanced by Gordo et al. (2017), where

a Siamese Network is trained with a Triplet loss on the Landmarks dataset (Babenko et al. 2014). However, Revaud et al. (2019a) argue 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 which 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 Chen et al. (2017a). This work is then followed-up with a flexible attention-based model for region extraction by Chen et al. (2018). Khaliq et al. (2019) 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 Jenicek and Chum (2019) to achieve illumination-invariance for place recognition.

Other Interesting Approaches to VPR: Other interesting approaches to place recognition include semantic-segmentation-based VPR (as in Arandjelović and Zisserman 2014b; Mousavian et al. 2015; Stenborg et al. 2018; Schönberger et al. 2018; Naseer et al. 2017) and object-proposals-based VPR (Hou et al. 2018), as recently reviewed by Garg et al. (2020). For images containing repetitive structures, Torii et al. (2013) proposed a robust mechanism for collecting visual words into descriptors. Synthetic views are utilized for enhanced illumination-invariant VPR in Torii et al. (2015), which shows that highly condition-variant images can still be matched, if they are from the same viewpoint. In addition to image retrieval, significant 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, see the survey by Kostavelis and Gasteratos (2015). Most of these semantic mapping techniques are based on Bayesian-surprise (Ranganathan 2013; Girdhar and Dudek 2010), coresets (Paul et al. 2014), region proposals (Demir and Bozma 2018), change-point detection (Topp and Christensen 2008; Ranganathan 2013) and salience-computation (Zaffar et al. 2018).

While the VPR literature consists of a large number of VPR techniques, we have currently implemented 8 state-of-the-art techniques into the VPR-Bench framework. We have also added the provision to integrate results (image descriptors) from other techniques, which has been demonstrated by integrating DenseVLAD and GeM into the benchmark. We plan to increase this number over time due to the modular nature of our framework with the help of the VPR community.

Benchmarks for Visual-localisation: Within the performance evaluation landscape, if we broaden our scope, it is evident that ours is not the first attempt at benchmarking visual-localisation at scale and previous attempts exist, which have led to the rapid development in this domain. From the computer vision perspective, the well-established visual-localisation benchmark[2] 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 has recently also included VPR (image-retrieval) benchmarking by combining with the Mapillary Street Level Sequences (MSLS) dataset (Warburg et al. 2020) 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 recent detailed works from Torii et al. (2019) and Sattler et al. (2018). Google also proposed the Landmarks dataset with focus on both place/instance-level recognition and retrieval: Google Landmark V1 dataset (Noh et al. 2017) and Google Landmark V2 dataset (Weyand et al. 2020). 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. Here, another divide is that of direct versus indirect evaluation of image retrieval, where the former directly quantifies the performance of a VPR system's output, while the latter assesses the performance of a larger system using end-task metrics such that VPR is only a module of this system's pipeline. The scope of VPR-Bench is limited to the direct evaluation of VPR.

Direct and Indirect Evaluation Metrics for VPR: 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 our work and 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 measuring the performance of the complete pipeline, where indirectly a good performing VPR module contributes to but

[2] www.visuallocalization.net.

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 Sturm et al. (2012). 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 Sattler et al. (2018).

Within direct performance evaluation, the most dominant VPR evaluation metric in robotics literature (Lowry et al. 2015) has been Area-under-the-Precision-Recall curves (denoted usually as AUC-PR or simply AUC), which tries to summarise the Precision-Recall curves in a single quantified value. AUC-PR favours techniques that can retrieve the correct match as the top ranked image, thus favouring applications that require highly precise localisation estimates. The reasons for 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, Lategahn et al. (2013), Cieslewski and Scaramuzza (2017), Ye et al. (2017), Camara and Přeučil (2019), Khaliq et al. (2019) and Tomită et al. (2021). Other than AUC-PR, F1-score has also been used in VPR evaluations predominantly by the robotics-focused VPR community, for example by Mishkin et al. (2015), Sünderhauf et al. (2015), Talbot et al. (2018), Garg et al. (2018b) and Hausler et al. (2019), 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 localisation every few meters, Porav et al. (2018) present a new metric/analysis to compute the VPR performance, using the maximum distance traversed by a robot without achieving a true-positive/localisation/loop-closure. Recently, Ferrarini et al. (2020) presented a new metric Extended Precision (EP) for VPR evaluation that is based on Precision@100% Recall and Recall@100% Precision. In our previous work (Zaffar et al. 2020), we 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. Perronnin et al. (2010), Torii et al. (2013), Arandjelović and Zisserman (2014a), Torii et al. (2015), Arandjelovic et al. (2016) and Uy and Lee (2018). 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 Jegou et al. (2008), Gordo et al. (2016), Sattler et al. (2016), Gordo et al. (2017), Revaud et al. (2019a) and Weyand et al. (2020). Other than these metrics, Recall@Reduced Precision has also been used as an evaluation metric (Tipaldi et al. 2013) for place recognition. For computational analysis, feature encoding time, descriptor matching time and descriptor size have been the key metrics for both the communities.

It is evident that a large number of evaluation metrics can be employed for assessing the performance 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 the above discussed distribution of metrics is not depicting 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 optimisation in SLAM. The discussion and analysis on evaluation metrics scales quickly in the dimension of the number of metrics discussed. To limit the scope of this work, we have only used AUC-PR, RecallRate@N, true-positive trajectory distribution, feature encoding time, descriptor matching time and feature descriptor size as our evaluation metrics in this work. We discuss these metrics systematically and at length later in Sect. 3.4.

Invariance Evaluation of VPR: The effect of viewpoint and appearance variations on visual place recognition has been well studied in the past, aiming to understand the limitations of different approaches. Chen et al. (2014b) and Sünderhauf et al. (2015) evaluated different convolutional layers of off-the-shelf CNNs for their performance on VPR and concluded that mid-level and higher-level layers were respectively more robust to appearance and viewpoint variations. Garg et al. (2018a) validated this trend on a more challenging scenario of opposing viewpoints while also showcasing catastrophic failure of viewpoint-dependent representations due to 180 degrees shift in camera viewpoint. In a subsequent work, Garg et al. (2018b) presented an empirical study on the amount of translational

offset needed to match places from opposing viewpoints in city-like environments. Pepperell et al. (2015) studied the effect of scale on VPR performance when using side-view imagery and travelling in different lanes within city suburbs and on a highway. Chéron (2018) evaluated the performance of local features for recognition using '*free viewpoint* videos' and concluded that traditional hand-crafted features demonstrated more viewpoint-robustness than their learnt counterparts. Kopitkov and Indelman (2018) characterized the viewpoint-dependency of CNN feature descriptors and used it to improve probabilistic inference of a robot's location. In this work, we present a more formal treatment to the effect of viewpoint and appearance variations on VPR by utilizing the Points Features dataset (Aanæs et al. 2012) for performance quantification. We then extend this analysis to real-world scenarios using the QUT Multi-Lane dataset (Skinner et al. 2016) and MIT Multi-Illumination dataset (Murmann et al. 2019).

Algorithm A Generic VPR System

Given: Q, M_R
Required: P, S, C

def *compute_query_desc* (Q)
  Preprocessing Steps
  Function Body
  Postprocessing Steps
  return F_Q

def *compute_map_features* (M_R)
  Preprocessing Steps
  Function Body
  Postprocessing Steps
  return F_M

def *perform_VPR* (F_Q, F_M)
  Preprocessing Steps
  Function Body
  Postprocessing Steps
  return P, S, C

def *main* ()
  F_M = *compute_map_features* (M_R)
  F_Q = *compute_query_desc* (Q)
  P, S, C = *perform_VPR* (F_Q, F_M)
  store P, S, C

3 VPR-Bench Framework

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

3.1 VPR Task Formulation

Here, we formally define what a VPR system represents throughout this paper.

Let Q be a query image and M_R be a list/map of R reference images. The feature descriptor(s) of a query image Q and reference map M_R can be denoted as F_Q and F_M, respectively. If a technique uses ROI-extraction, F_Q will hold within it all the required information in this regards, including location of regions, their descriptors and corresponding salience. The input Q can also be a sequence of Query images and any other pre/post-processed form of a query candidate. For a query image Q, given a reference map M_R, let us denote the best matched image/place by a VPR technique as P (where, $P \in M_R$) with a matching score S. The matching score S can be defined as $S \in [0, 1]$. The confusion matrix (matching scores with all reference images) can be denoted as C. Based on these notations, the following algorithm represent a VPR system.

3.2 Evaluation Datasets

In this section, we present the existing patterns and features of datasets in VPR and then discuss each of the datasets that have been used in this work by dividing them into outdoor and indoor datasets categories.

3.2.1 Dataset Considerations in VPR-Bench

All the datasets that have been employed to date for VPR evaluation comprise of 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 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 Sect. 4.6. 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 m apart may have significant visual overlap due to high scene depth, while frames that are 5 m 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 image-matching 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, we have taken the following steps:

1. We have integrated datasets that contain a large variation in the acceptable ground-truth viewpoint variance: ranging from the minimally acceptable viewpoint variation in the Corridor dataset to the large acceptable viewpoint variations of the Tokyo 24/7 dataset, thus to cover a broader audience.
2. We have provided an extensive analysis on the effects of changing acceptable levels of viewpoint invariance in Sect. 4.6.
3. As for consistency in VPR research and performance reporting, it is essential to affix a unified template for all of these VPR datasets, we will be re-releasing all datasets in a VPR-Bench compatible mode with their associated ground-truth information.

Despite the extensive collection of datasets in this work, 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. We have designed VPR-Bench as per unified templates to allow integration of new datasets. Further details of the datasets template are provided in the appendix of this paper.

3.2.2 Outdoor Environment

We have integrated multiple outdoor datasets in our 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. Each of these datasets has a particular attribute to offer, that lead to its selection and they are briefly discussed below.

The GardensPoint dataset was created by Glover (2014) and first used for VPR by Chen et al. (2014b), 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. We have only used the day and night sequences in our work because they contain both the viewpoint and conditional change. The Tokyo 24/7 dataset was proposed by Torii et al. (2015), which consists of 3D viewpoint-variations and time-of-day variations. We use version 2 of the query images, as suggested by the authors of Torii et al. (2015) and Arandjelovic et al. (2016) 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. (2018) and is the only dataset designed with focus on perceptual aliasing and confusing places/frames for VPR techniques. The SPEDTest dataset was introduced by Chen et al. (2018) 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 Ros et al. (2016) and represents a simulated city-like environment in various weather, seasonal and time of day conditions. In this paper, we have 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 our work represents a traversal from Larsson et al. (2019), which is a subset of the Oxford RobotCar dataset (Maddern et al. 2017). 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 Skrede (2013) and introduced to VPR evaluation by Sünderhauf et al. (2013), which represents a 728 km of train journey in Norway during Summer and Winter seasons. As Nordland

Table 1 The 12 VPR-Bench datasets integrated into VPR-Bench and used in this study are enlisted here

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	75,984	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	$\tilde{3}20 \times \tilde{2}40$	$\tilde{3}20 \times \tilde{2}40$
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	27,592	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	23,000	3D	None	640 × 480	640 × 480
INRIA Holidays	Outdoor	300	512	Lateral/3D	None	$\tilde{2}50 \times \tilde{1}85$	$\tilde{2}50 \times \tilde{1}85$

The sign $\tilde{}$ for image resolutions (pixels × pixels) indicates datasets where image resolution varies in-between different images of the dataset and we have therefore specified the common resolution observed in that dataset

dataset represents natural (non-urban), outdoor environment, which is unexplored in any other dataset, we have integrated it into VPR-Bench. From the computer vision community, in addition to Tokyo 24/7, we have used the Pittsburgh dataset (Torii et al. 2013) and the INRIA Holidays dataset (Jegou et al. 2008) to bridge the important gap between the two communities. We use only the query images of Pittsburgh dataset because this represents the only large-scale dataset in our 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.

3.2.3 Indoor Environment

A significant focus in recent research in VPR has primarily been on evaluation on outdoor datasets, so we also incorporate indoor environments 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. We have briefly discussed the currently available indoor datasets in VPR-Bench, in the following paragraph.

We have integrated the 17-Places dataset introduced by Sahdev and Tsotsos (2016) 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. We also use the viewpoint-variant Corridor dataset, introduced by Milford (2013), which represents the challenge of low-resolution and feature-less images (160 × 120 pixels) for vision-based place recognition. Mount and Milford (2016) 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. This dataset only contain 32 queries and 32 references, we deliberately use such a small-scale dataset to see the ordering of VPR techniques on very small-scale datasets.

3.2.4 Ground-Truth Information

Because we have 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. We have 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 will be re-released and therefore we have only briefly presented this ground-truth information in Table 2. The ground-truth tolerance for some of the robotics-focused

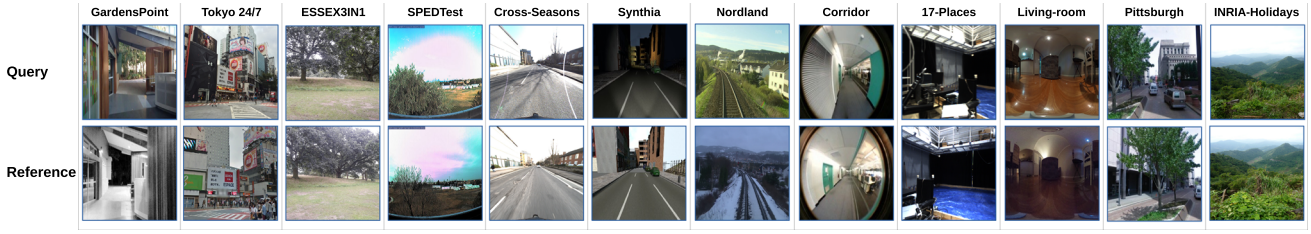

Fig. 3 Sample images from all 12 VPR datasets employed in this work are presented here. These datasets span many different environments, including cities, natural scenery, train-lines, rooms, offices, corridors, buildings, busy-streets and such

Table 2 The ground-truth tolerance for the 12 VPR-Bench datasets integrated into VPR-Bench is provided here

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

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

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, we have used the original levels being used by their respective communities.

3.3 VPR Techniques

In this section, we introduce the 10 VPR techniques that have been evaluated in this work, while also providing important implementation details of these techniques that are needed to understand the experiments and results in the next Sect. 4.

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 our framework, based upon its performance as shown by McManus et al. (2014) and the value it presents as an underlying feature descriptor for training a convolutional auto-encoder in Merrill and Huang (2018). We use 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 are set equal to 9. We use cosine-matching between HOG-descriptors of various images to find the best place match.

AlexNet: The use of AlexNet for VPR was studied by Sünderhauf et al. (2015), 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. Our implementation of AlexNet is similar to the one employed by Merrill and Huang (2018), while the code has been restructured as per the designed template. Note that AlexNet resizes input image to 227 × 227 before it is input to the neural network.

DenseVLAD: DenseVLAD has been proposed by Torii et al. (2015), 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. In this work, we have formatted (as per our template) and integrated the descriptor matching data computed by the DenseVLAD code open-sourced by Torii et al. (2015) into VPR-Bench to demonstrate the utility of our framework for cases where code conversion may not be required/desired. All input images are resized to 640 × 480, similar to Torii et al. (2015).

AP-GeM: GeM was originally proposed by Radenović et al. (2018), 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 Revaud et al. (2019a), where they have designed a new ranking-loss based on mean-Average-Precision. We have used the GeM code open-sourced by Revaud et al. (2019a) based on the ResNet101 model (namely ResNet101-AP-GeM) with an output descriptor size of 2048 dimensions. Similar to DenseVLAD, we have used the descriptor matching data computed by the original code of the respective authors and integrated that with our framework for a seamlessly straightforward integration process. Revaud et al. (2019a) 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, we resized input images to 640 × 480.

NetVLAD: The original implementation of NetVLAD was in MATLAB, as released by Arandjelovic et al. (2016). The Python port of this code was open-sourced by Cieslewski et al. (2018). The model selected for evaluation is VGG-16, which has been trained in an end-to-end manner on Pittsburgh 30K dataset (Arandjelovic et al. 2016) with a dictionary size of 64 while performing whitening on the final descriptors. The code has been modified as per our template. The authors of NetVLAD have suggested an image resolution of 640 × 480 at inference time and we have therefore used this image resolution for all experiments.

AMOSNet: This technique was proposed by Chen et al. (2017b), 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. We have implemented AMOSNet as per our 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 implementation of AMOSNet resizes input images to 227 × 227.

HybridNet: While AMOSNet was trained from scratch, Chen et al. (2017b) took inspiration from transfer learning for HybridNet and re-trained the weights initialised from Top-5 convolutional layers of CaffeNet (Krizhevsky et al. 2012) on SPED dataset. We have implemented HybridNet as per our 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. (2019). We have modified it as per our template and have used AlexNet trained as Places365 dataset as the underlying CNN. The total number of ROIs has been set to 400 and we have used 'conv3' 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 implementation of RegionVLAD resizes input images to 227 × 227.

CALC: The use of convolutional auto-encoders for VPR was proposed by Merrill and Huang (2018), 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. We use model parameters from 100,000 training iteration and adapt the open-source technique as per our template. Cosine-matching is used for descriptor comparison. This is the only semi-supervised learning technique in our 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 (Zaffar et al. 2020) 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 our template. We have used an image-size of 512 × 512, cell-size of 16 × 16, bin-size of 8 and an entropy-threshold (ET) of 0.4. CoHOG also uses cosine-matching for descriptor comparison.

3.4 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 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. We have 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.

We have collated the taxonomy of various metrics used in VPR by both the computer vision and the robotics communities in Table 3 for the reader's reference, which are also discussed later in the paper. 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

Table 3 A taxonomy of VPR evaluation metrics is given here

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

PL primary localisation, *LC* loop-closure, *IR* image retrieval, *FP* false-positives, *RC* robotics community, *CV* computer vision community, *MB* matching-based, *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

respective metric. We have broadly classified the usage 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, we discuss each of the metrics that have been used for evaluations in this work, their motivation and limitations.

3.4.1 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, 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 value, 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 backends) 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

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}, \tag{1}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}. \tag{2}$$

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 (we also discuss this later in the paper 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.

3.4.2 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 has 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 consistency 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, we have included this metric into our framework.

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. Both the metrics neither 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

$$RecallRate@N = \frac{M_Q}{N_Q}. \tag{3}$$

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.

3.4.3 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, we utilise the well-established Receiver-Operating-Characteristic (ROC) curve. 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 due to which 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. We therefore manually add true-negatives to the Gardens Point dataset for our ROC evaluation, where true-negatives are images taken from the Nordland dataset as a case-study. The modified Gardens Point dataset contain the 200 original true-positives and the added 200 true-negatives from 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 our framework to avoid confusion with the original datasets. It is easily possible to extend this analysis on other datasets and is supported by our 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 labelled as false-positives. These metrics are computed as

$$TPR = \frac{True\ Positives}{True\ Positives + False\ Negatives}, \tag{4}$$

$$FPR = \frac{False\ Positives}{False\ Positives + True\ Negatives}. \tag{5}$$

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 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.

3.4.4 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,[3] descriptor matching time may be more relevant than feature encoding time and vice-versa.

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. We model this t_R as

$$t_R = t_e + O(Z) \times t_m. \tag{6}$$

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, we 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 work for completeness.

Additional to the metrics discussed previously, we also compute and report the feature descriptor size of all VPR techniques to reflect the storage requirements, which are highly relevant for large-scale maps.

3.4.5 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.. We have therefore included true-positives distribution over trajectory analysis in our 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 Porav et al. (2018). They analyse the distribution of loop-closure candidates (true-positives) by creating histograms identifying inter-loop-closure distances, such that the height of the his-

[3] 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.

togram bar specifies the number of loop-closures performed in the dataset with that particular inter-frame distance constraint. We use the same analysis schema in this work.

3.4.6 Other VPR Metrics

The metrics discussed previously in this paper 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 (Revaud et al. 2019a), Performance-per-Compute-Unit (Zaffar et al. 2020; Tomită et al. 2020), Recall@0.95 Precision (Chen et al. 2011), Extended Precision (Ferrarini et al. 2020), F1-score (Hausler et al. 2019), error-rate (Chen et al. 2014a) and Recall@100% Precision (Chen et al. 2014b). To limit the scope of the analysis performed in this paper, and because there is a high corelation between some of these metrics (e.g. between RecallRate@N, Recall@100% Precision and Recall@95% Precision), we have implemented many of these other metrics in the implementation of VPR-Bench, but did not include them in this paper.

3.5 Invariance Quantification Setup

In this sub-section (and its respective results/analysis in Sect. 4.8) we propose a thorough sweep over a wide range of quantified viewpoint and illumination variations and study the effect on VPR techniques.

Aanæs et al. (2012) proposed a well-designed and highly-detailed dataset, namely 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, we utilise a subset of the dataset that represents scenes of synthetically-created 'Places', and we use 2 of these scenes/places in our work. We have integrated this subset of the Point Features dataset in our framework and Sect. 3.5.1 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, we do make an effort in this paper 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 our evaluation to some degree. Therefore, in this reference, we have used the QUT multi-lane dataset (Skinner et al. 2016) for viewpoint variations and the MIT multi-illumination dataset (Murmann et al. 2019) for illumination variations. Details of both of these datasets are available in their respective sub-sections below.

We have also dedicated a sub-section (Sect. 3.5.4) to present the details of our evaluation mechanism on these 3 datasets. The evaluation mechanism in this paper (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 section 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'.

3.5.1 Point Features Dataset

The Point Features dataset can be broadly classified to have 3 variations: (1) Viewpoint, (2) Illumination and (3) Scene. We fully use the former two variations in our work, while only two relevant scenes (representing two different places) are utilised from the latter. The authors Aanæs et al. (2012) 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. 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. 6, while the azimuth (ϕ) and elevation angle (θ) of each LED is listed in Table 4. Figure 4 shows various components of the dataset, while in Fig. 7 we qualitatively show all the 19 different illumination cases on one of the scenes.

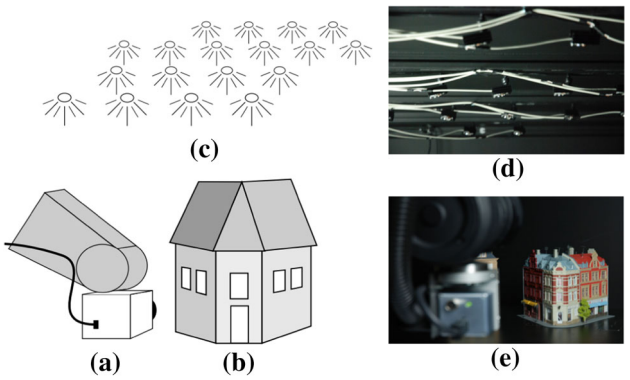

Fig. 4 The schematic setup of the Point Features dataset has been reproduced here with permission from Aanæs et al. (2012). The dataset primarily consists of **a** A camera mounted on a robot-arm, **b** Synthetic Scene, **c** LED arrays for illumination, **d, e** Snapshots of the actual setup

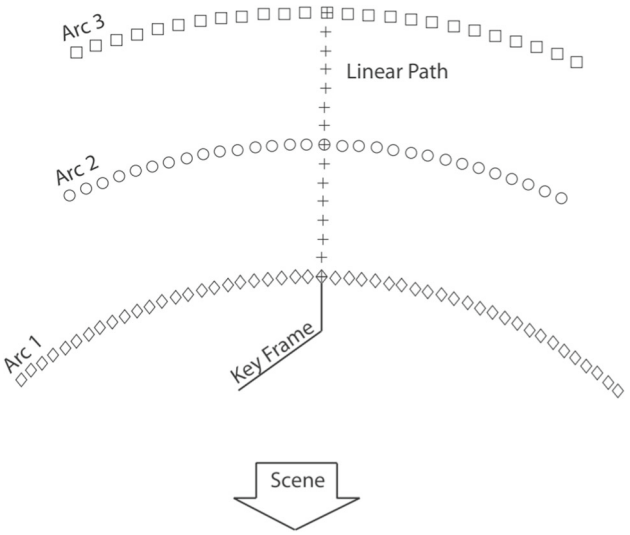

Fig. 5 The 119 different viewpoints in the Point Features dataset have been reproduced here with permission from Aanæs et al. (2012). Camera is directed towards the scene from all viewpoints. Arc 1, 2 and 3 span 40, 25 and 20 degrees, respectively, while the radii are 0.5, 0.65 and 0.8 m

3.5.2 QUT Multi-lane Dataset

The QUT multi-lane dataset is a small-scale dataset depicting a traversal through an outdoor environment (Skinner et al. 2016) 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.

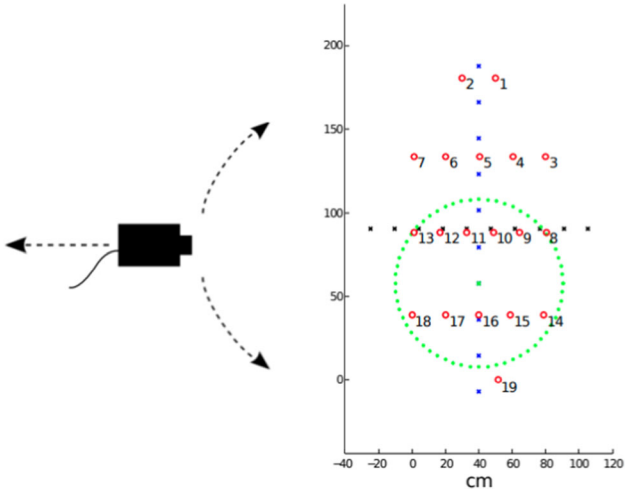

Fig. 6 The distribution of LEDs across physical space is shown as seen from above. Each red circle represents an LED and only a single LED is illuminated at a point in time, yielding 19 different illumination conditions. In the original work, Aanæs et al. (2012), used artificial linear relighting from left-to-right (blue) and front-to-back (black) based on a Gaussian-weighting, as depicted with the green-circle, but in our work we have only used the original 19 single-LED illuminated cases. These 19 cases (red-circles) need to be seen in correspondence with Table (Color figure online) 4

Table 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			

We use in total 2 different scenes (representing 2 different places) from their traversal and for each scene use 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 m, while the forward/backward viewpoint change is around 3.5 m. Examples of these viewpoint changes have been shown in Fig. 8 for both the scenes/places.

Fig. 7 The change in appearance of a scene for 19 different illumination levels is shown here from the Point Features dataset

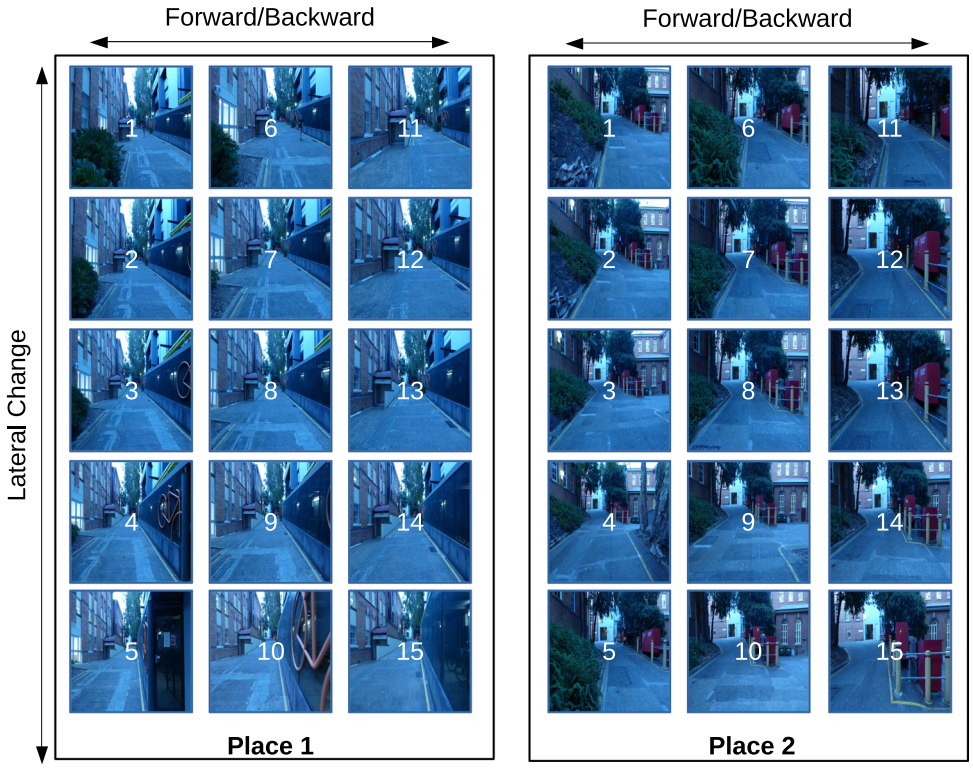

Fig. 8 The 15 different viewpoint cases in the QUT multi-lane dataset for both the scenes/places have been presented here

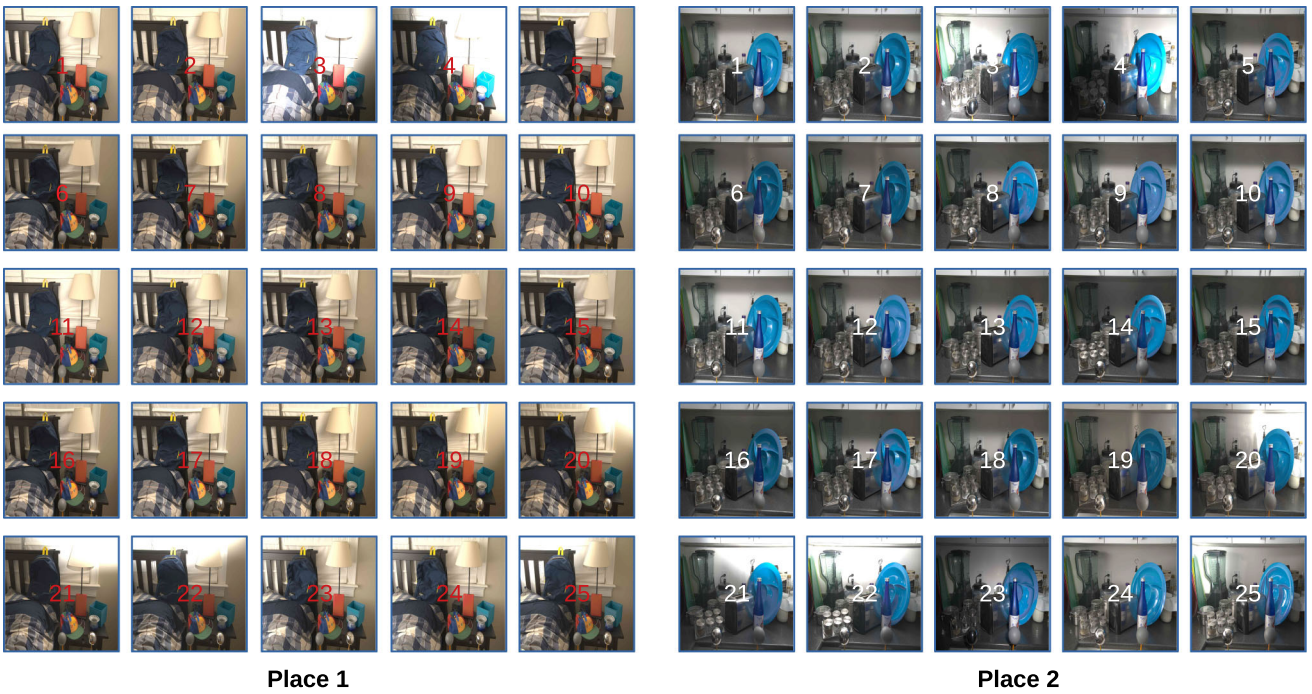

Fig. 9 The 25 different illumination cases for both the scenes/places from the MIT multi-illumination dataset have been presented here

3.5.3 MIT Multi-illumination Dataset

The MIT multi-illumination dataset was recently proposed by Murmann et al. (2019). 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 we only require 2 scenes/places, we have manually mined scenes that represent an indoor appearance of a place and are feature-full.[4]

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 Mohan et al. (2007) and Murmann et al. (2016). 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 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 and a plastic gray sphere, at the bottom edge of every image. For further specification details, we would refer the reader to the original paper of Murmann et al. (2019) for avoiding textual redundancies. Examples of the 2 different places under the varying illumination conditions have been shown in Fig. 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.

3.5.4 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), we had to devise an analysis scheme where VPR performance variation could be quantified and analysed. 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

[4] The authors acknowledge that even the multi-illumination dataset may not fully represent a real-world 'landmark' and multiple illumination sources etc, however to the best of authors' 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 Murmann et al. (2019).

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 no. 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 fully-immune to such variations and a useful analysis would be to see this effect on the confidence-score.

Therefore, our analysis on the 3 datasets in this section 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, we draw 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.

Evaluation Mechanism Point Features Dataset: There are a total of 119 different viewpoint positions and 19 different illumination levels. We consider the illumination case 1 in Fig. 6 and the left-most point on Arc 1 of Fig. 5 as our 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 right-most 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, we keep the illumination type/level constant, move along Arc 1 in a clock-wise fashion and compute 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 4 and qualitatively in Fig. 7. For the illumination-invariance analysis, the viewpoint position is kept constant (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, we draw the matching scores for the same-but-varied scene in the Point Features dataset, 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, we utilise all of the 119 viewpoint positions and the 19 illumination levels of the different scene/place. This gives us the corresponding number (119/19 for both variations) of data-points for the confidence scores between keyframe and the different place to be drawn 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 Sect. 4.8.

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 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. 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/utilised 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. 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

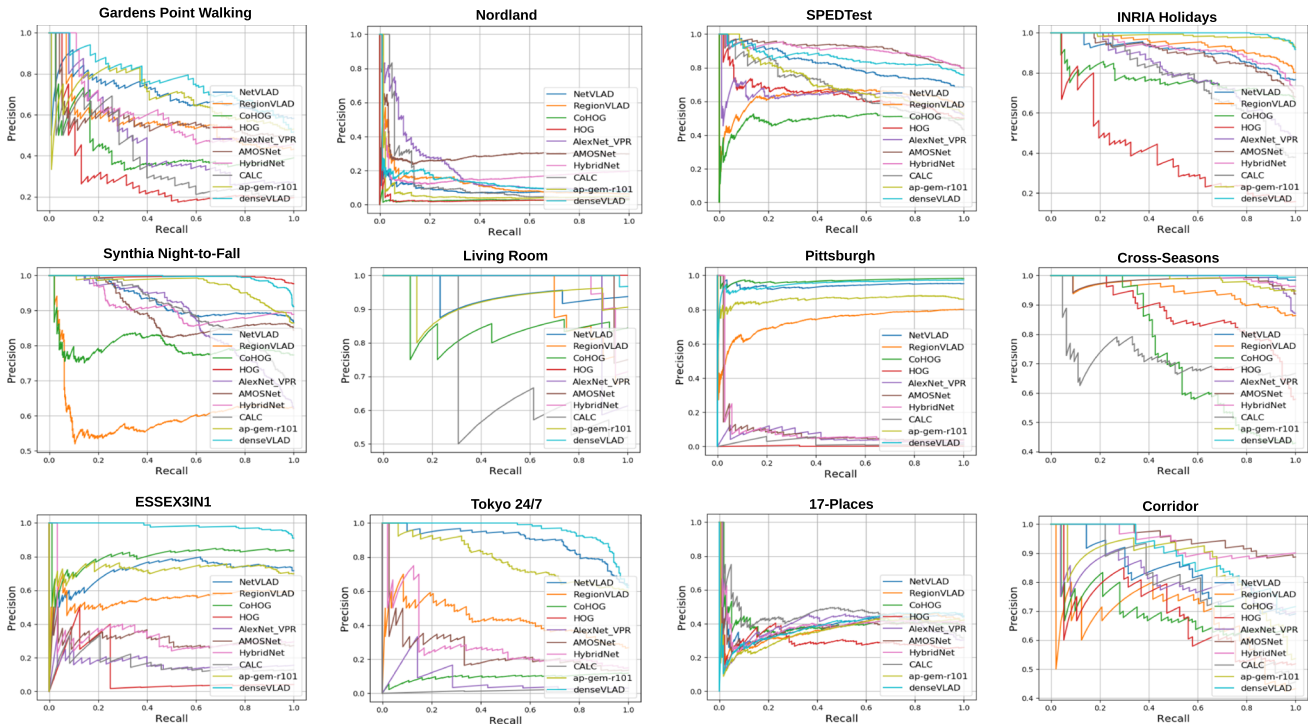

Fig. 10 The Precision-Recall curves for all 10 VPR techniques generated on the 12 datasets by VPR-Bench framework are presented here

keyframe and the 25 different illuminations of different place (place 2) are computed/utilised in the same fashion as that for the Point Features dataset.

4 Results and Analysis

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

4.1 Place Matching Performance

We now present the results obtained by executing the VPR-Bench framework given the attributes presented in Sect. 3.

PR-Curves: Firstly, the precision-recall curves for all 10 VPR techniques on the 12 indoor and outdoor datasets are presented in Fig. 10. The values of AUC-PR for all techniques have been listed in Table 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 Room dataset, which consists of high-quality images of places under indoor illu-

Table 5 The values of AUC-PR are listed here for all the techniques on the 12 datasets

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**

The bold values in each row represent the state-of-the-art technique for each dataset for the corresponding metric

mination 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. RegionVLAD 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 5, RecallRate@N is usually represented as a trend and not as a single value. Therefore, for RecallRate@N, we plot the variations in RecallRate for values of N in the range of 1 to 20. These plots have been created for all the 10 VPR techniques on the 12 datasets and are shown in Fig. 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. 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 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. 12 for a qualitative insight.

Computational Performance: The values of feature encoding time, descriptor matching time and descriptor size have been listed in Table 6 for our fixed platform. For all experiments in this work, we have used the default data-types of descriptors as specified in Table 6 last row, however for the sake of complete comparison of matching time t_m, we affixed data-type of all techniques to float-64 for the values of t_m in Table 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 versus HOG in Table 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

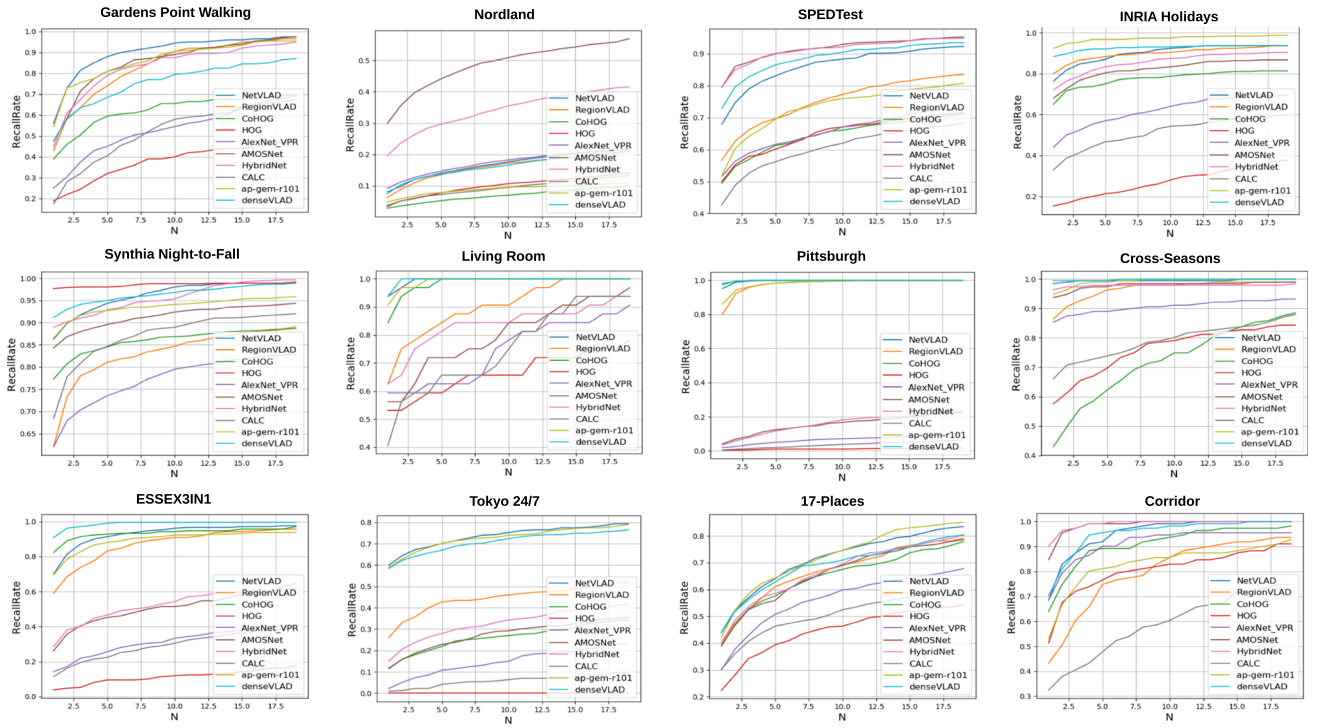

Fig. 11 The RecallRate@N curves for all 10 VPR techniques generated on the 12 datasets by VPR-Bench framework are presented here. The range of N used here is 1 to 20 with a step-size of 1. The values of RecallRate@1 represent the Precision@100% Recall of a VPR technique

versus CoHOG in Table 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 versus HOG in Table 6 third row). AMOSNet has half the descriptor size than CoHOG, both descriptors are 2-dimensional, but the matching time for CoHOG is significantly higher than AMOSNet due to different distance functions, i.e. L1-matching for AMOSNet and cosine-distance for CoHOG.

Some of the key findings from the analysis in this subsection can be summarised as follows:

1. Unlike previous evaluations (Zaffar et al. 2019a, b), where state-of-the-art AUC-PR performance was almost always achieved by NetVLAD, this paper 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 Sattler et al. (2018) and Torii et al. (2019), and we re-affirm their findings here. In our work, we also show how HOG and CoHOG have achieved state-of-the-art performance for all metrics on at least one dataset (see results on Synthia Night-to-Fall dataset and Pittsburgh dataset in Table 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 hand-crafted computationally-efficient technique, as suggested by results in Table 5 for 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. 11 and Table 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 retrieving 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

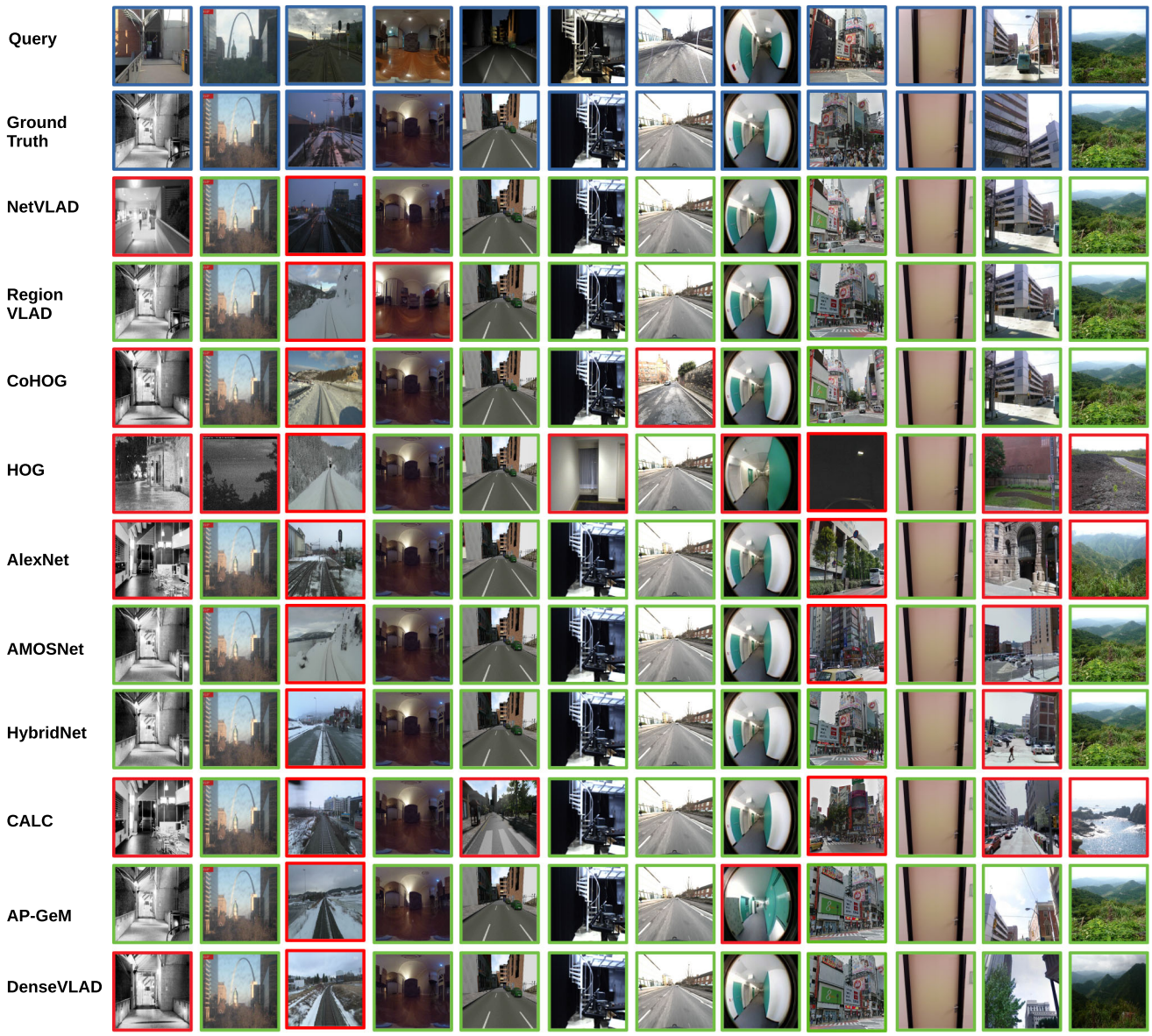

Fig. 12 Exemplar images matched/mismatched by VPR techniques are shown here for a qualitative insight. Red bounded images are incorrect matches (false positives) and green-bounded images are correct matches (true positives). An image is taken from each of the 12 datasets, where the order of datasets from left to right follows the same sequence as top to bottom in Table 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 matching performance of a VPR-system (Color figure online)

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 run-time memory (RAM) consumption/needs. We further present analysis on this in Sect. 4.4.

Table 6 The values of feature encoding time t_e (sec), descriptor matching time t_m (msec) are listed here for 8 VPR techniques

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	$1 \times 34{,}596$	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

Encoding time is dependent upon the image resolution, however in this work we 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 fixed float-64 data-type of descriptors for all techniques. Please see accompanying text regarding trends of 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

4.2 ROC Curves: Finding New Places

Next, we show the ROC curves for all techniques on a modified version of the Gardens Point dataset. We have modified the Gardens Point dataset 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. 13. We note 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 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).

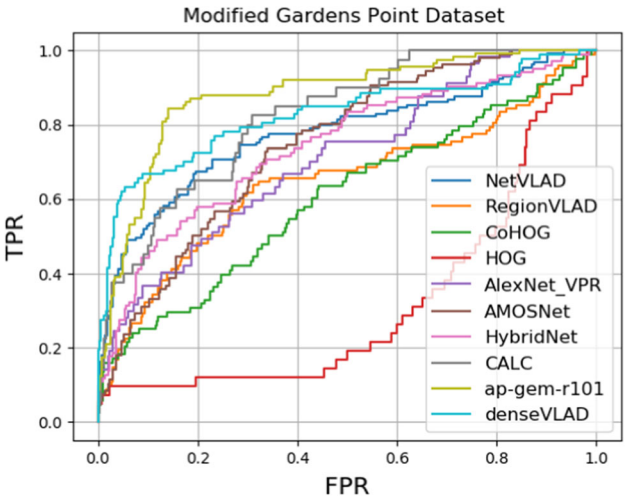

Fig. 13 The ROC performance of 10 VPR techniques is shown here on a modified (true-negative added) version of Gardens Point dataset that contains 200 true-negatives and 200 true-positives

4.3 Computational Performance: CPU versus 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 versus GPU), it may or may not be fair to compare handcrafted VPR techniques with deep-learning-based VPR techniques on computational front.

We here report the feature encoding time t_e and the descriptor matching time t_m of the 7 deep-learning-based techniques in our suite when implemented on a GPU-based platform. The GPU-based evaluation was performed using an Nvidia GeForce GTX 1080 Ti with 12 GB 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. We have reported these timings in Table 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 6), with notable exception of RegionVLAD versus NetVLAD for t_e, because of the former's compute-intensive CPU-based

Table 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

region-extraction and VLAD description. In general, the computation times between CPU and GPU vary noticeably for all the methods. This cross-analysis highlights the varying utility of VPR techniques across different platforms.

4.4 Descriptor Size Analysis

In this sub-section, we further extend upon the descriptor size analysis and show 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, we use the Gardens Point dataset and change various descriptor-related parameters of 5 VPR techniques, namely CoHOG, HOG, NetVLAD, DenseVLAD and AP-GeM, that directly affect the descriptor size.

For HOG and CoHOG, we have changed the cell-size of the HOG-computation scheme, 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, we changed the PCA output dimensions 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 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 the decrease of 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 9), decrease of descriptor size may or may not lead to 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. (2020), where increasing cell-size leads to reduced viewpoint invariance and lesser AUC-PR. For HOG the 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.

Table 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 (s)	t_m (ms)
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**

The bold values represent the state-of-the-art

4.5 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 Sect. 3.4.5, we report in Fig. 14 the distribution of true-positives for 6 trajectory-based datasets. The distribution here refers to no. of true-positives (Y-axis) for a given distance (X-axis) between two correctly retrieved frames. For all the datasets, we have assumed an inter-frame distance of 1 m, i.e. true-positives that are assumed to be 5 m apart represent two correctly-matched query frames that are 5 frames apart. This assumption is required because we do not have the exact knowledge of inter-frame physical distance for all

Table 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

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
8×8	0.47	508	47.0	8×8	0.19	571	0.14	4096	0.69	16.30	0.06	4096	0.77	16.30	0.06	4096	–	–	
16×16	0.42	123	2.64	16×16	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
32×32	0.36	28.8	0.18	32×32	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
64×64	0.30	6.27	0.06	64×64	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
128×128	0.19	1.15	0.05	128×128	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
256×256	0.12	0.128	0.03	256×256	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

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

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. 14. For most techniques on all the datasets, the loop-closures are distributed evenly i.e. curves in Fig. 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 (Cummins and Newman 2011) which rely solely on location estimated through VPR. We have not provided this analysis for non-trajectory-type datasets (SPEDTest, INRIA Holidays etc), because the inter-frame distance is not a valid assumption for these cases.

4.6 Acceptable Ground-Truth Manipulation

An important finding from the analysis performed for Sect. 4.1 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. We show this in Fig. 15 for the Nordland and Gardens Point dataset. Here we use a small section of the Nordland traversal (as used in Merrill and Huang 2018; Zaffar et al. 2019b) 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 in 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. 16. We report this effect of changing ground-truth range on the AUC-PR of various VPR techniques for the Gardens Point dataset and Nordland dataset in Fig. 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 Sect. 4.9. 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.

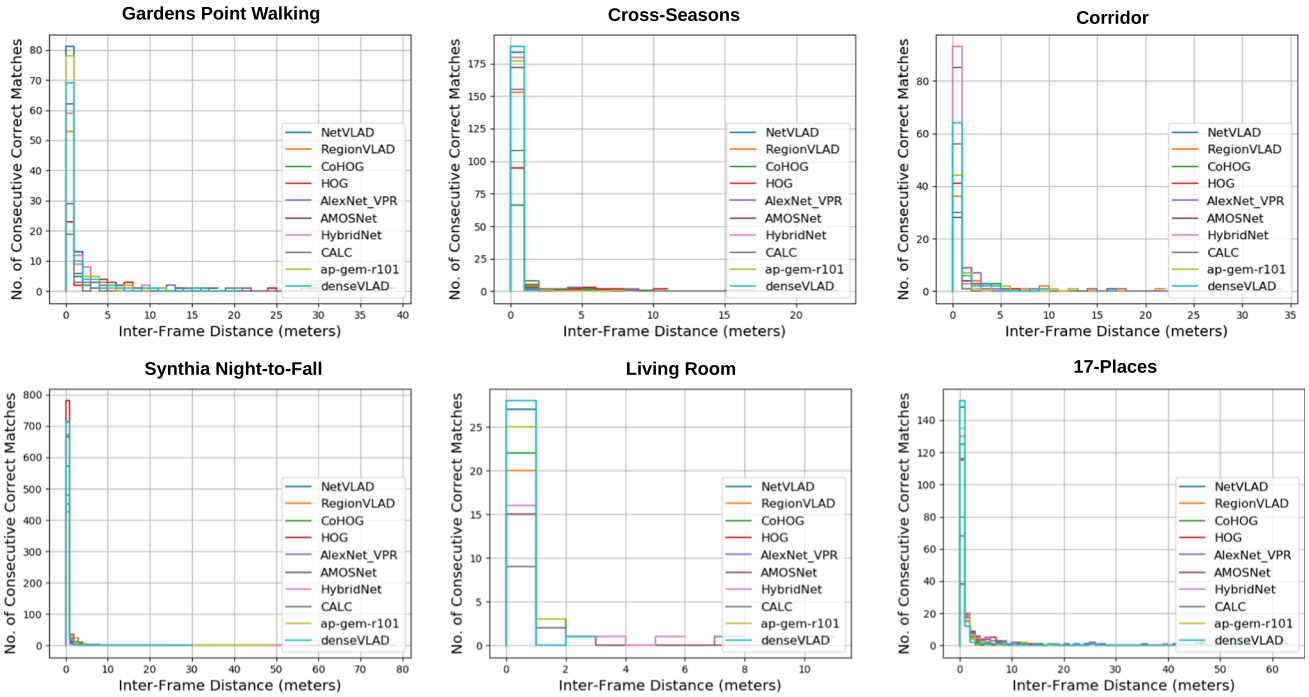

Fig. 14 The distribution of true-positives over the trajectory of a dataset are shown here. The horizontal axis represents the distance between two consecutive true-positives in a sequence and the vertical axis shows the number of true-positives that satisfy this distance constraint

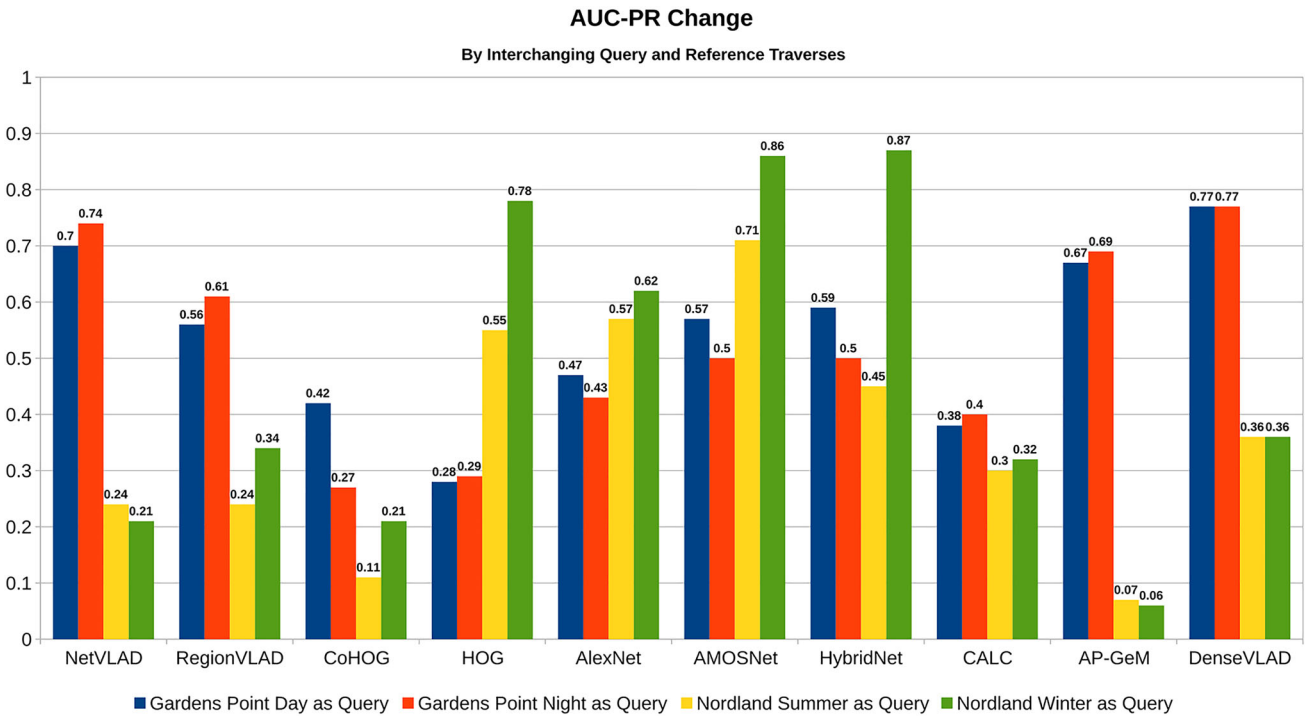

Fig. 15 The effect on AUC-PR performance of techniques by inter-changing the query and reference traverses is shown here for the Gardens Point dataset and Nordland dataset

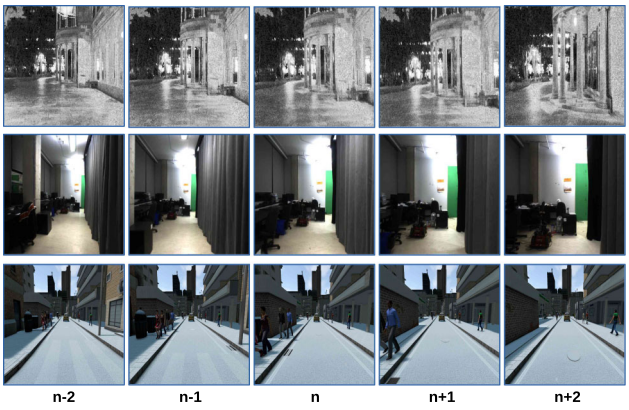

Fig. 16 The overlap between visual information among subsequent images in traversal-based datasets is shown here. Depending on what level of ground-truth true positive range is acceptable, benefits will be distributed among the techniques based on their viewpoint-invariance

4.7 Retrieval Time versus Platform Speed

One of the questions that we wanted to address through this manuscript is, 'What is a good image-retrieval time?'. This is important because most VPR research papers (as covered in our literature review) that claim real-time performance consider anything between 5–25 frames-per-second (FPS) as real-time. However, there are 2 important caveats to such performance. Firstly, the retrieval performance for a VPR application depends on the size of the map. It is therefore very 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, while a dynamic platform traverses an environment. In such a case, the utility of a technique will depend upon the speed of the platform, as the faster the platform moves, the lower the retrieval time that is acceptable. We have modelled this as follows.

Let us assume that a particular application requires K frames-per-meter (where K could be fractional) and that the platform moves with a velocity V. Also, let the size of the map (no. 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

$$FPS_{req} = K \times V. \tag{7}$$

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 Eq. (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), we draw the retrieval performance of all techniques along the platform speed for different values of Z in Fig. 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.

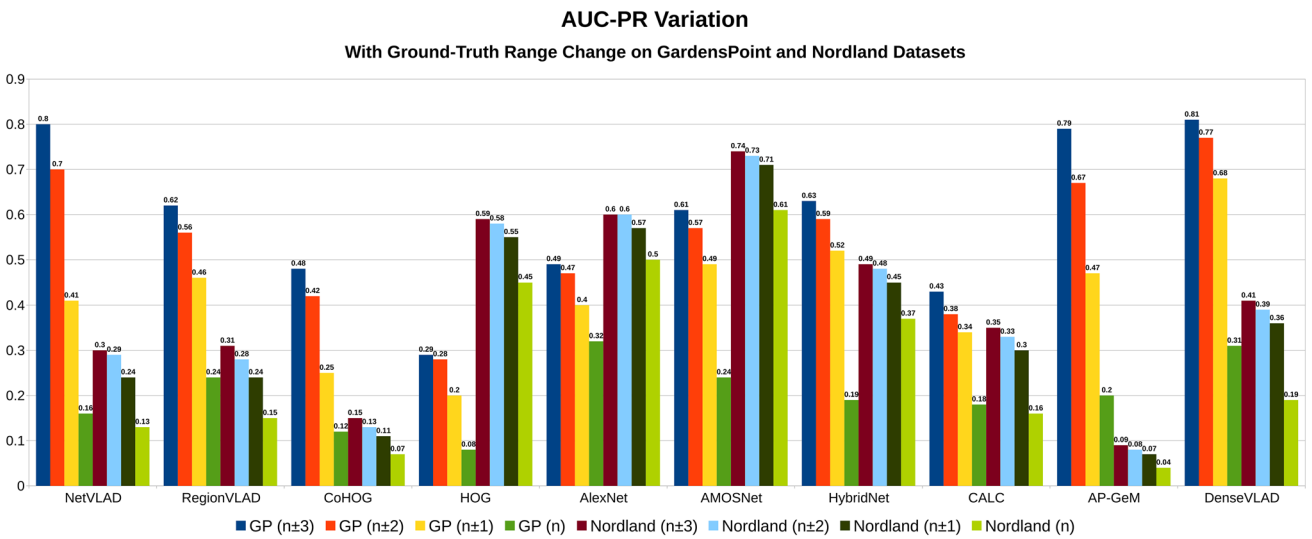

Fig. 17 The effect on AUC-PR performance of techniques by changing the range of ground-truth true positive images is shown here for the Gardens Point dataset and Nordland dataset

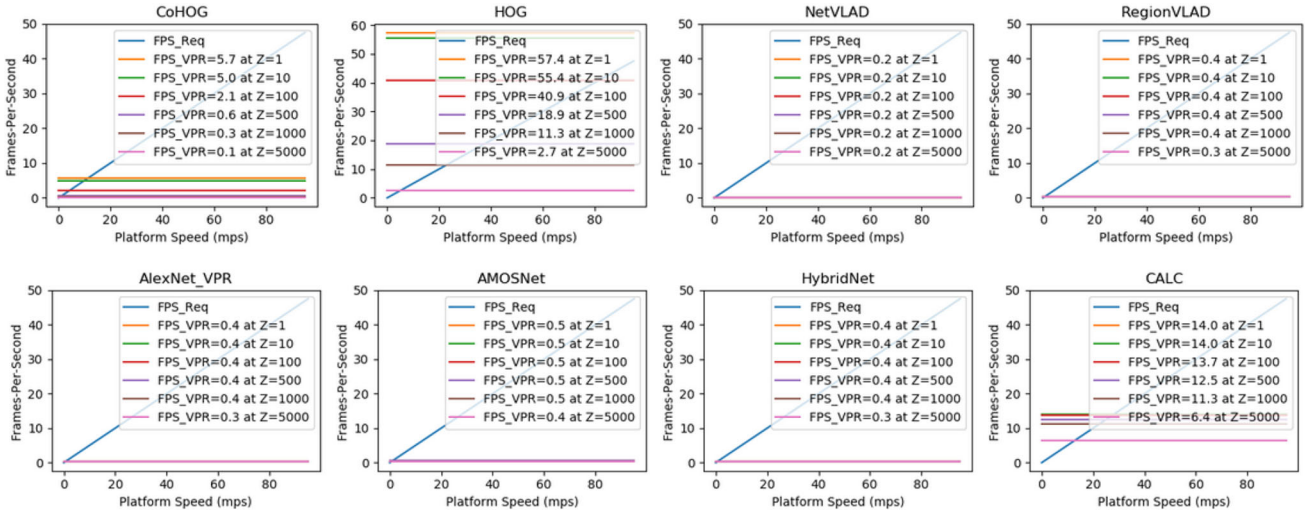

Fig. 18 The retrieval performance of techniques is drawn for different map-sizes (Z) across the platform speed. Depending upon the value of frames required per meter (K) for an application, these curves will scale linearly according to Eq. (7)

4.8 Invariance Analysis

One of the key aspects of the VPR-Bench framework as explained in Sect. 3 is the quantification of viewpoint- and illumination-invariance of a VPR technique. In Sect. 4.1, we had utilised the traditional VPR analysis schema, where datasets are usually classified based on the qualitative severity of a particular variation. However, in this section, we utilise the Point Features dataset presented in Sect. 3.5 and utilise the quantitative information presented in Figs. 5, 6 and Table 4.

The change in matching score along these arcs is shown in Fig. 19 for all the techniques. There is 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-appearance 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 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 work is immune to all levels of viewpoint-variation.

Another benefit of having the matching scores curves for different place 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. 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 in only based on two places/scenes.

We have extended the analysis of viewpoint-invariance from the synthetic Point Features dataset to the real-world QUT Multi-lane dataset. The analysis scheme is the same for both the datasets and the obtained curves are shown in Fig. 20. The curves on the QUT Multi-lane dataset re-affirm our findings from the Point Features dataset and the trends on both the 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

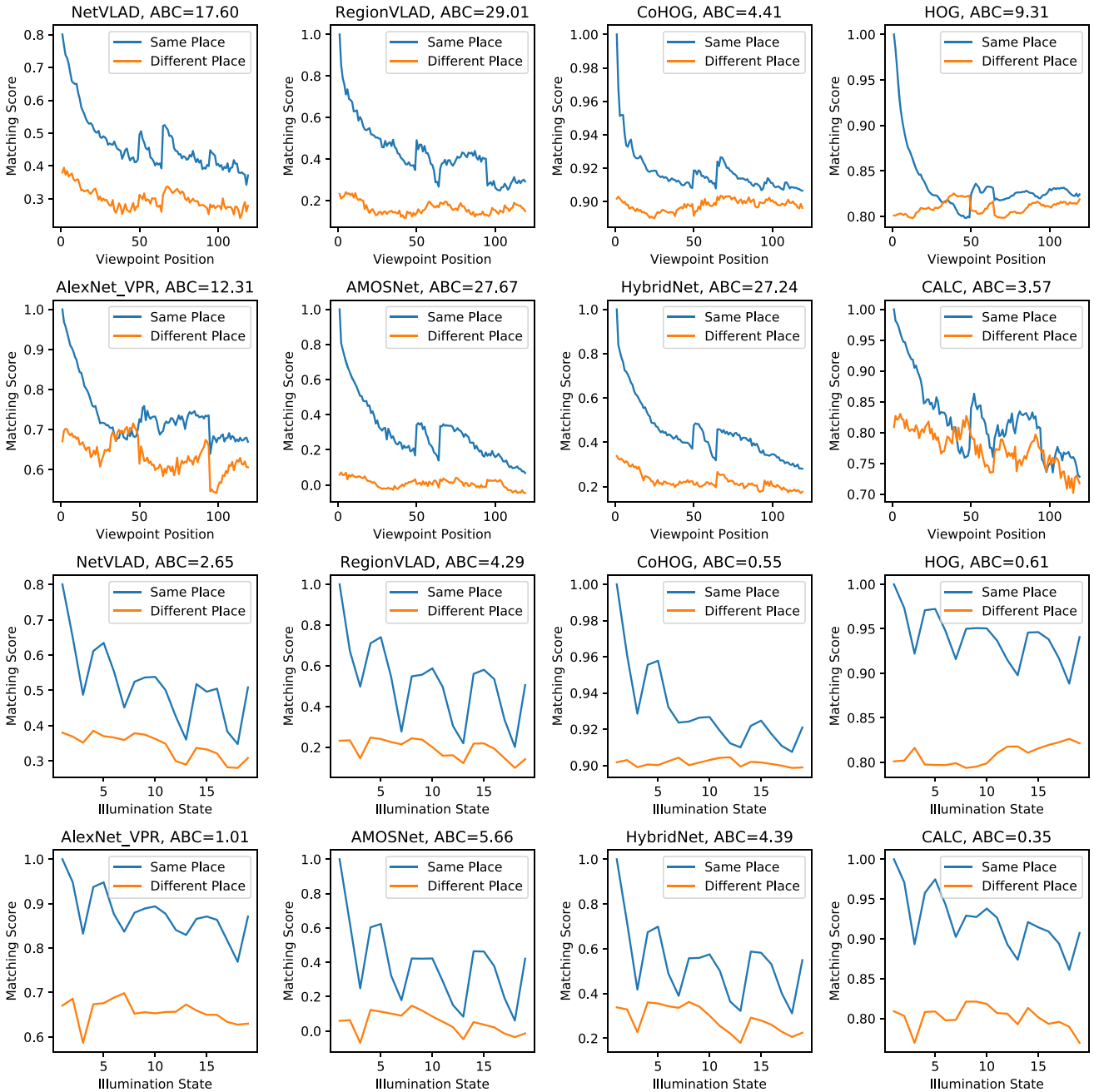

Fig. 19 The change in matching score for quantified viewpoint and illumination variations is shown here on the Point Features dataset. The first two rows contain changes for all techniques with 119 viewpoint positions, while the bottom two row show these changes for 19 different illumination levels.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

viewpoint variations on both Point Features dataset and 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. 19. While the 119 different viewpoint positions represented in Fig. 5 are intuitive for analysis, the nature and level of illumination change in Table 4 is not obvious. We have presented these 19 different cases qualitatively in

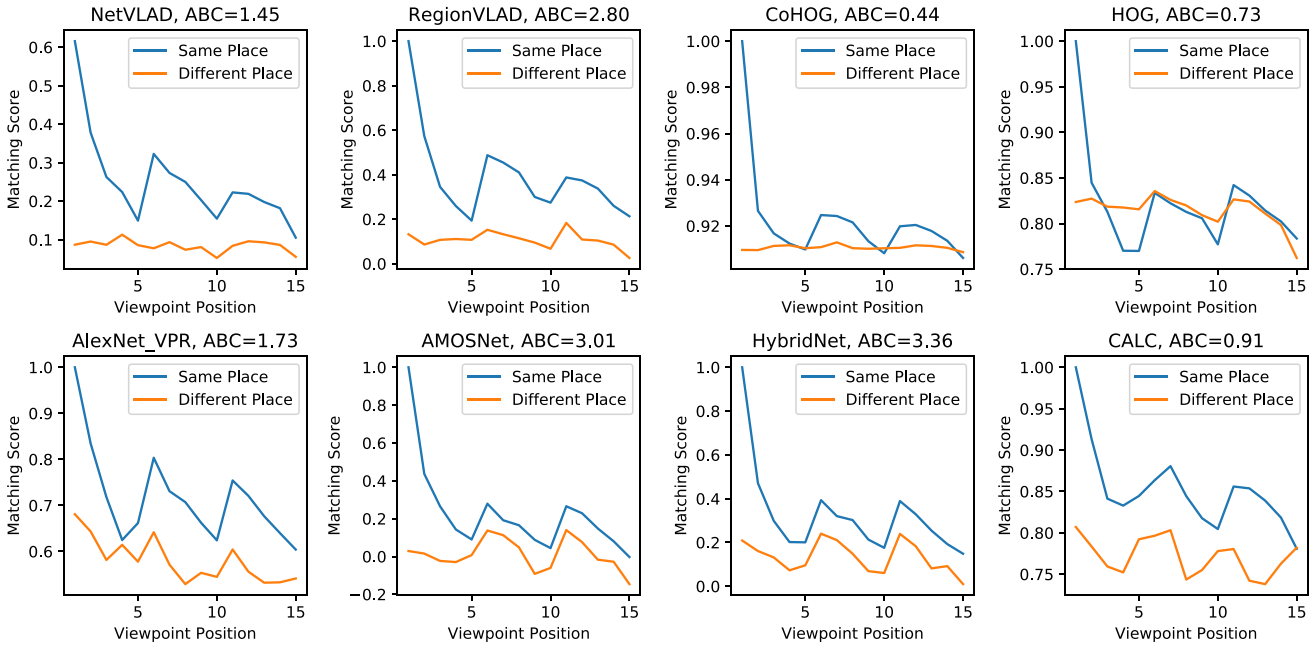

Fig. 20 The change in matching score for the quantified viewpoint variations is shown here on the QUT Multi-lane dataset. The confidence score variation is shown for all techniques against the 15 viewpoint positions, as explained in Sect. 3.5.4

Fig. 7, so that the illumination-variance curves in Fig. 19 can be easily 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 has 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.

We have extended our illumination-invariance analysis from the Point Features dataset to the MIT Multi-illumination dataset and the curves on Multi-illumination dataset are presented in Fig. 21. There is a very sharp drop in place matching confidence for illumination cases 3 and 4 for all the VPR techniques, which re-affirms our finding on the Point Features dataset regarding the significantly large effect of directional illumination change (see Fig. 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 generalise to other learning-based techniques. The reported performance decline by varying illumination cases can be potentially combined with illumination-source prediction works (Gardner et al. 2017; Hold-Geoffroy et al. 2017) to predict when a VPR technique might fail and how different VPR techniques could complement each other in these scenarios.

4.9 Variance versus 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, we take 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 are the applications where VPR techniques act as the primary localisation module and where, there is no image-to-image, epipolar-geometry-based motion estimation (location refinement) module. For example, Zeng et al. (2019) extend the concept of VPR for precise localisation in mining environments. Similar extensions of VPR as the only module for precise-localisation 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 which 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 2 images of the same place are viewpoint-varied, the VPR technique can distinguish between them to perform metrically-precise

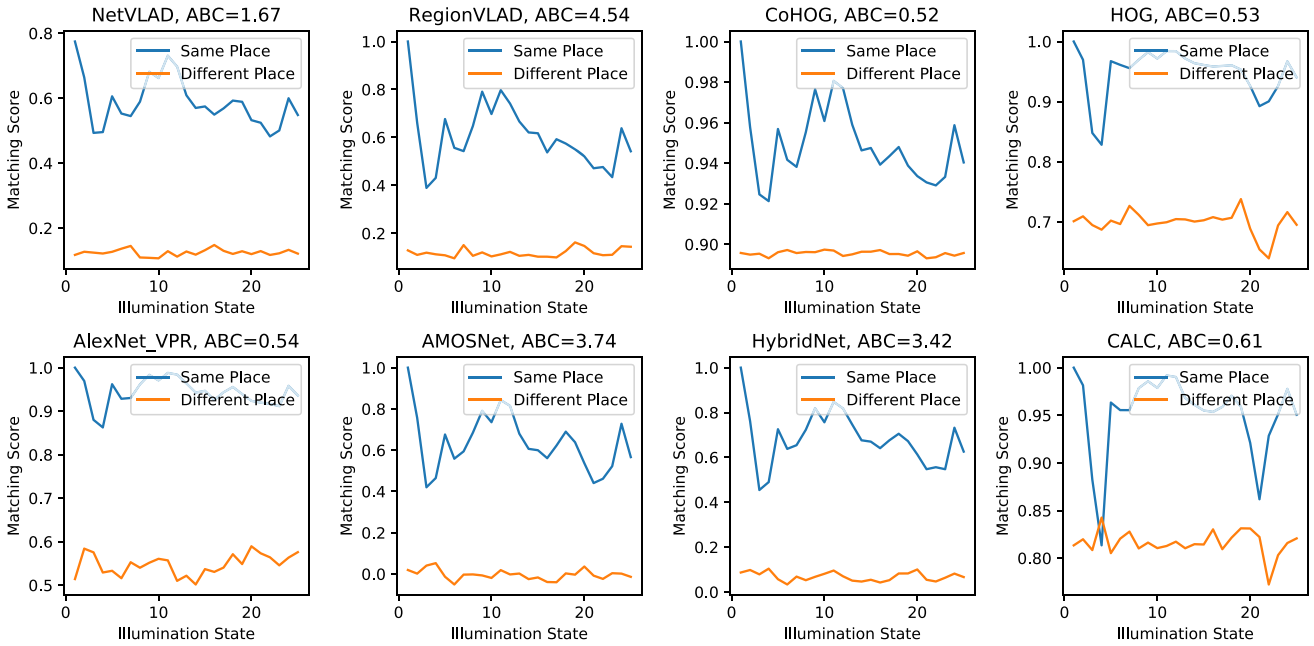

Fig. 21 The change in matching score for the illumination variations is shown here on the MIT Multi-illumination dataset. The confidence score variation is given for all techniques on the 25 illumination positions, as explained in Sect. 3.5.4

localisation. If a viewpoint-invariant technique is utilised in this scenario, the inherent viewpoint-invariance will lead to discrepancies in localisation estimates and eventually cause a system failure.

Thus, a key area to investigate within VPR research should be controlled viewpoint-variance. In Sect. 4.8, we 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. We believe that this is an exciting research challenge and should be a topic for VPR research in the upcoming years. Nevertheless, our proposal is that both viewpoint-variance and invariance are desirable properties, depending upon the underlying application and should be regarded/investigated accordingly.

5 Conclusions and Future Work

In this paper, we presented a comprehensive and variation-quantified evaluation framework for visual place recognition performance. Our open-source framework VPR-Bench integrates 12 different indoor and outdoor datasets, along with 10 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. We utilised the variation- and illumination-quantified Point Features dataset to evaluate and analyse the level and nature of variations that a VPR technique can handle. We then extended this analysis and our findings from the synthetic Point Features dataset to the QUT Multi-lane dataset and the MIT multi-illumination dataset.

Using our framework, we provide a number of useful insights about the nature of challenges that a particular technique can handle. We identify that no universal state-of-the-art technique exists for place matching and discuss the reasons behind the success/failure of these techniques from one dataset to another. In our 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 specialised techniques and pipelines for VPR. We also report 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. Our 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. Our 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 which cannot benefit much from a GPU in terms of encoding time due to its CPU-bound intense region-extraction scheme. We demonstrate 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 investigating the trade-offs between reduced matching time at reduced descriptor precision and size. Overall, our work 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.

Our new analysis for viewpoint and illumination-invariance quantification is developed around the Point Features dataset, and integrated within the framework for ease-of-use by other VPR researchers. Our results on this dataset identify that 3D viewpoint change has 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. We further show that directional illumination change presents a bigger challenge for VPR than uniform illumination change, both for deep-learning and handcrafted techniques. We also propose that viewpoint variance instead of viewpoint invariance can also be important for VPR systems, e.g. for accurate localisation, sensitivity to viewpoint change can be a feature. Because we have employed a number of different datasets, techniques and metrics, VPR-Bench enables many more performance comparisons, and we have only discussed a few selected comparisons to limit the scope.

It remains future work to further investigate the relation between place matching performance and the bottle-necks 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 work, but also deserves more detailed future investigation. It would also be useful to include evaluations on more challenging environments, such as under-water 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. We hope that this work proves useful for both the computer vision and the robotics VPR communities to compare newly proposed techniques in detail to the state-of-the-art on these varied datasets using diverse evaluation metrics. We are keen on integrating more VPR techniques into VPR-Bench and encourage any feedback, collaborations and suggestions.

Acknowledgements Our work was supported by the UK Engineering and Physical Sciences Research Council through Grants EP/R02572X/1, EP/P017487/1 and EP/R026173/1 and in part by the RICE project funded by the National Centre for Nuclear Robotics Flexible Partnership Fund. This research has also been supported by the TU Delft AI Labs programme. Michael Milford was supported by ARC Grants FT140101229, CE140100016 and the QUT Centre for Robotics.

Appendices

A VPR-Bench Design

A.1 Code Structure

The entire framework has been designed with 2 key focuses: (a) A holistic, fully-integrated and easy-to-use framework for VPR performance evaluation at all fronts, (b) Modularity and convenient templates for regular updates and future consistency. In this respect, while the modularity, template design and available content within the modules, are explained individually for each of the modules in their respective dedicated sub-sections; this sub-section presents the overall framework structure and implementation details. The code structure of our framework has been described in Fig. 22.

The entry to the framework is a convenient main file, where the choice of evaluation datasets, VPR techniques and evaluation mode can be specified. At present there are 2 evaluation modes: (1) VPR Performance Evaluation and (2) Invariance Analysis. The former yields the place matching performance of different VPR techniques (implemented within the framework and/or integrated using pre-computed matching information) on a specified dataset using different metrics related to precision and computation. The latter tries to present the invariance of these techniques to quantified viewpoint- and illumination-variations. There are 12 evaluation datasets available in the framework from both indoor and outdoor environments. We have re-implemented 8 different VPR techniques by modifying the open-source codes as per our templates or self-implementing in cases where open-

source codes were not available. The VPR-Bench framework is written fully in Python (2.7) (working on upgrading to Python 3), which has been the most used programming/scripting language for VPR research. Our framework does not have a dedicated Graphical-User-Interface (GUI), because the framework is targeted for developers/researchers who are assumed to have basic knowledge of the domain. GUIs also make future improvements much complex and limit the flexibility of an application. The open-source code has been tested on a Ubuntu 20.04.1 LTS system. By default, the framework does not need a GPU (Graphical Processing Unit) for any of the evaluations. Therefore, a huge percentage of VPR researchers, academics and developers, from a broad range of application domains can conveniently use our framework.

A.2 Integrating New Datasets and Techniques

As we are focused on providing flexibility and ease for integrating new VPR techniques and datasets into VPR-Bench (additional to the already available 12 datasets and 8 techniques), we have briefly summarised the required steps for both of these changes below:

1. For integrating a new dataset into VPR-Bench, no change in the framework code is required. You need to setup the dataset as per our unified template, which has been explained in Appendix B. and then set the directory path for this dataset in the main file.
2. There are two possible ways to integrate a new VPR technique into VPR-Bench: (a) Re-implement the technique as per our template within the VPR-Bench framework, (b) Use pre-computed data through an external implementation of the technique. We encourage the former, where the main file for this respective technique needs to implement 3 functions, as per the template described in Appendix C. Once these functions have been implemented, they only need to be imported in our framework and all other modules will be implicitly integrated for this technique. The benefit of re-implementing a technique as per our template is the ease for new researchers to understand, utilize and modify the implementation of these various VPR techniques based on a fixed and compact template. Moreover, templates also make computational analysis more fair, by affixing the input and output pipelines (i.e. the time taken to input and output data to a VPR techniques' various functions). For the latter, we maintain a provision in our framework to re-use pre-computed data through an external implementation and integrate it with the features offered by our framework. The computational analysis for techniques integrated via external implementations (non-template) is still relevant (albeit will vary based upon the implementation) as long as the underlying hardware is the same. A unified template has been developed for integrating pre-computed data, that takes in the matching scores for all the query images with all the reference images, feature encoding time and descriptor matching time. We have integrated DenseVLAD and GeM using this pre-computed data in our work. The details for integrating VPR techniques in this fashion will also be provided in the files supporting the release of our open-source code.

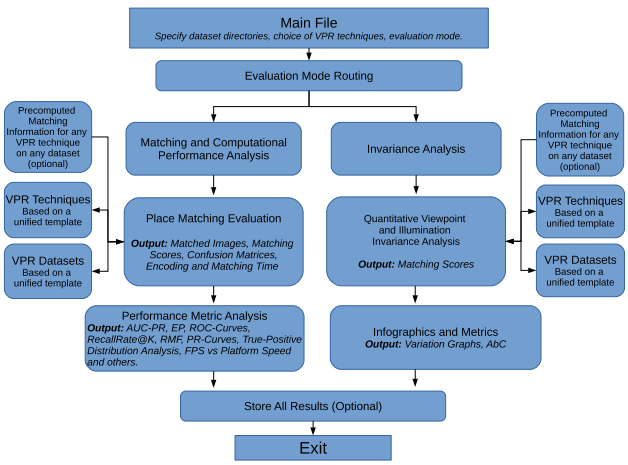

Fig. 22 The code structure of the VPR-Bench framework is shown here

B VPR-Bench Datasets Template

In order to have a fixed template for all the datasets that are available in (or can be integrated into) VPR-Bench, we design a simplistic, generic template that can accommodate the variations within the dataset formats. Firstly, the query and reference traverses for a dataset are represented by their dedicated sub-folders. Images within each of these folders need to be named as integers, which is motivated by a graph structure, such that for a traversal-based dataset, increments or decrements to integer values can represent the temporally and/or geographically next or previous image, respectively. The ground-truth file for each dataset is a numpy array (.npy). This multi-dimensional numpy array of ground-truth information has dimensions of $T_Q \times 2$, where T_Q is the total number of query images in the dataset. For all T_Q rows of query images, the first column represents the query image index and the second column contains the list of indices of all ground-truth matching reference images. We have used the simplistic image indices/names as our choice of ground-truth, because they can be parsed from a range of different modalities, like GPS information, pose-information and/or manual frame correspondences, as shown in this work by restructuring all the 12 datasets to the described common template.

C VPR-Bench Techniques Template

Each VPR technique has a different approach to the problem, which may include neural-network models or traditional feature descriptors. There may be added functionality, like ROI-extraction, image pre-processing, descriptor adaptation, usage of sequential and/or geometric prior etc. The designed templates for techniques have the provision to allow for such pre- and post-processing steps. We also provide a parallel path in our pipeline to seamlessly integrate pre-computed place matching information from a different technique running on the same/different platform.

Let Q be a query image and M_R be a list/map of R reference images. The feature descriptor(s) of a query image Q and reference map M_R can be denoted as F_Q and F_M, respectively. If a technique uses ROI-extraction, F_Q will hold within it all the required information in this regards, including location of regions, their descriptors and corresponding salience as a multi-dimensional list. The input Q can also be a sequence of Query images and any other pre/post-processed form of a query candidate. For a query image Q, given a reference map M_R, let us denote the best matched image/place by a VPR technique as P (where, $P \in M_R$) with a matching score S. The matching score S can be defined as $S \in [0, 1]$. The confusion matrix (matching scores with all reference images) can be denoted as C. Based on these notations, the following 3 functions need to be implemented in the main file of a VPR technique. The definitions (names) of these functions remain the same for all VPR techniques and our framework performs technique-aware selective re-imports of these functions to maintain consistency and ease-of-integration.

Algorithm VPR Technique Required Template

```
def compute_query_desc (Q)
    Preprocessing Steps
    Function Body
    Postprocessing Steps
    return F_Q
def compute_map_features (M_R)
    Preprocessing Steps
    Function Body
    Postprocessing Steps
    return F_M
def perform_VPR (F_Q, F_M)
    Preprocessing Steps
    Function Body
    Postprocessing Steps
    return P, S, C
```

References

Aanæs, H., Dahl, A. L., & Pedersen, K. S. (2012). Interesting interest points. *International Journal of Computer Vision*, *97*(1), 18–35.

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., et al. (2011). Building Rome in a day. *Communications of the ACM*, *54*(10), 105–112.

Agrawal, M., Konolige, K., & Blas, M. R. (2008). Censure: Center surround extremas for realtime feature detection and matching. In *European conference on computer vision* (pp. 102–115). Springer.

Andreasson, H., & Duckett, T. (2004). Topological localization for mobile robots using omni-directional vision and local features. *IFAC Proceedings Volumes*, *37*(8), 36–41.

Angeli, A., Doncieux, S., Meyer, J. A., & Filliat, D. (2008). Incremental vision-based topological slam. In *IROS* (pp. 1031–1036) IEEE.

Arandjelović, R., & Zisserman, A. (2014a). Dislocation: Scalable descriptor distinctiveness for location recognition. In *Asian conference on computer vision* (pp. 188–204). Springer.

Arandjelović, R., & Zisserman, A. (2014b). Visual vocabulary with a semantic twist. In *Asian conference on computer vision* (pp. 178–195). Springer.

Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., & Sivic, J. (2016). NetVLAD: CNN architecture for weakly supervised place recognition. In *CVPR* (pp. 5297–5307).

Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014). Neural codes for image retrieval. In *European conference on computer vision* (pp. 584–599). Springer

Badino, H., Huber, D., & Kanade, T. (2012). Real-time topometric localization. In *ICRA* (pp. 1635–1642). IEEE.

Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. In *ECCV* (pp. 404–417). Springer.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., et al. (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE T-RO*, *32*(6), 1309–1332.

Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., & Fua, P. (2011). Brief: Computing a local binary descriptor very fast. *IEEE T-PAMI*, *34*(7), 1281–1298.

Camara, L. G., Gäbert, C., & Preucil, L. (2019). Highly robust visual place recognition through spatial matching of CNN features. ResearchGate Preprint.

Camara, L. G., & Přeučil, L. (2019). Spatio-semantic convnet-based visual place recognition. In *2019 European conference on mobile robots (ECMR)* (pp. 1–8). IEEE.

Cao, B., Araujo, A., & Sim, J. (2020). Unifying deep local and global features for image search. arXiv:2001.05027

Chancán, M., Hernandez-Nunez, L., Narendra, A., Barron, A. B., & Milford, M. (2020). A hybrid compact neural architecture for visual place recognition. *IEEE Robotics and Automation Letters*, *5*(2), 993–1000.

Chen, D. M., Baatz, G., Köser, K., Tsai, S. S., Vedantham, R., Pylvänäinen, T., Roimela, K., Chen, X., Bach, J., Pollefeys, M., et al. (2011). City-scale landmark identification on mobile devices. In *CVPR 2011* (pp. 737–744).

Chen, Z., Jacobson, A., Erdem, U. M., Hasselmo, M. E., & Milford, M. (2014a). Multi-scale bio-inspired place recognition. In *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE

Chen, Z., Lam, O., Jacobson, A., & Milford, M. (2014b). Convolutional neural network-based place recognition. preprint arXiv:1411.1509.

Chen, Z., Maffra, F., Sa, I., & Chli, M. (2017a). Only look once, mining distinctive landmarks from convnet for visual place recognition. In *IROS* (pp. 9–16). IEEE.

Chen, Z., Liu, L., Sa, I., Ge, Z., & Chli, M. (2018). Learning context flexible attention model for long-term visual place recognition. *IEEE Robotics and Automation Letters*, *3*(4), 4015–4022.

Chen, Z., et al. (2017b). Deep learning features at scale for visual place recognition. In *ICRA* (pp. 3223–3230). IEEE.

Chéron, C. T. E. (2018). An evaluation of features for pose estimation and its application to free viewpoint video. PhD thesis, Trinity College.

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

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

Cummins, M., & Newman, P. (2011). Appearance-only slam at large scale with fab-map 2.0. *IJRR*, *30*(9), 1100–1123.

Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: Real-time single camera slam. *IEEE Transactions on Pattern analysis and Machine Intelligence*, *29*(6), 1052–1067.

Demir, M., & Bozma, H. I. (2018). Automated place detection based on coherent segments. In *2018 IEEE 12th international conference on semantic computing (ICSC)* (pp. 71–76). IEEE.

DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018). Superpoint: Self-supervised interest point detection and description. In *CVPR workshops* (pp. 224–236).

Dusmanu, M., et al. (2019). D2-net: A trainable CNN for joint description and detection of local features. In *CVPR* (pp. 8092–8101).

Ferrarini, B., Waheed, M., Waheed, S., Ehsan, S., Milford, M. J., & McDonald-Maier, K. D. (2020). Exploring performance bounds of visual place recognition using extended precision. *IEEE Robotics and Automation Letters*, *5*(2), 1688–1695.

Filliat, D. (2007). A visual bag of words method for interactive qualitative localization and mapping. In *ICRA* (pp. 3921–3926). IEEE.

Fraundorfer, F., Engels, C., & Nistér, D. (2007). Topological mapping, localization and navigation using image collections. In *2007 IEEE/RSJ international conference on intelligent robots and systems* (pp. 3872–3877). IEEE.

Gardner, M. A., Sunkavalli, K., Yumer, E., Shen, X., Gambaretto, E., Gagné, C., et al. (2017). Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics (TOG)*, *36*(6), 1–14.

Garg, S., Fischer, T., & Milford, M. (2021). Where is your place, visual place recognition? arXiv preprint arXiv:2103.06443.

Garg, S., Suenderhauf, N., & Milford, M. (2018a). Don't look back: Robustifying place categorization for viewpoint- and condition-invariant place recognition. In *IEEE international conference on robotics and automation (ICRA)*.

Garg, S., Suenderhauf, N., & Milford, M. (2018b). Lost? appearance-invariant place recognition for opposite viewpoints using visual semantics. In *Proceedings of robotics: Science and systems XIV*.

Garg, S., Sünderhauf, N., Dayoub, F., Morrison, D., Cosgun, A., Carneiro, G., et al. (2020). Semantics for robotic mapping, perception and interaction: A survey. *Found Trends Robot*, *8*(1–2), 1–224. https://doi.org/10.1561/2300000059.

Girdhar, Y., & Dudek, G. (2010). Online navigation summaries. In *2010 IEEE international conference on robotics and automation* (pp 5035–5040). IEEE.

Glover, A. (2014). Day and night, left and right. https://doi.org/10.5281/zenodo.4590133

Gordo, A., Almazán, J., Revaud, J.,& Larlus, D. (2016). Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*. (pp 241–257). Springer.

Gordo, A., Almazan, J., Revaud, J., & Larlus, D. (2017). End-to-end learning of deep visual representations for image retrieval. *International Journal of Computer Vision*, *124*(2), 237–254.

Hausler, S., Jacobson, A., & Milford, M. (2019). Multi-process fusion: Visual place recognition using multiple image processing methods. *IEEE Robotics and Automation Letters*, *4*(2), 1924–1931.

Ho, K. L., & Newman, P. (2007). Detecting loop closure with scene sequences. *IJCV*, *74*(3), 261–286.

Hold-Geoffroy, Y., Sunkavalli, K., Hadap, S., Gambaretto, E., & Lalonde, J. F. (2017). Deep outdoor illumination estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 7312–7321).

Hou, Y., Zhang, H., & Zhou, S. (2018). Evaluation of object proposals and convnet features for landmark-based visual place recognition. *Journal of Intelligent & Robotic Systems*, *92*(3–4), 505–520.

Jegou, H., Douze, M., & Schmid, C. (2008). Hamming embedding and weak geometric consistency for large scale image search. In *European conference on computer vision* (pp. 304–317). Springer.

Jégou, H., Douze, M., Schmid, C., & Pérez, P. (2010). Aggregating local descriptors into a compact image representation. In *CVPR* (pp. 3304–3311). IEEE Computer Society.

Jenicek, T., & Chum, O. (2019). No fear of the dark: Image retrieval under varying illumination conditions. In *Proceedings of the IEEE international conference on computer vision* (pp. 9696–9704).

Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K. M., & Trulls, E. (2020). Image matching across wide baselines: From paper to practice. arXiv preprint arXiv:2003.01587.

Johns, E., & Yang, G. Z. (2011). From images to scenes: Compressing an image cluster into a single scene model for place recognition. In *2011 International conference on computer vision* (pp 874–881). IEEE.

Khaliq, A., Ehsan, S., Chen, Z., Milford, M., & McDonald-Maier, K. (2019). A holistic visual place recognition approach using lightweight CNNs for significant viewpoint and appearance changes. *IEEE Transactions on Robotics*.

Konolige, K., & Agrawal, M. (2008). FrameSLAM: From bundle adjustment to real-time visual mapping. *IEEE Transactions on Robotics*, *24*(5), 1066–1077.

Kopitkov, D., & Indelman, V. (2018). Bayesian information recovery from cnn for probabilistic inference. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 7795–7802). IEEE.

Košecká, J., Li, F., & Yang, X. (2005). Global localization and relative positioning based on scale-invariant keypoints. *Robotics and Autonomous Systems*, *52*(1), 27–38.

Kostavelis, I., & Gasteratos, A. (2015). Semantic mapping for mobile robotics tasks: A survey. *RAS*, *66*, 86–103.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

Larsson, M., Stenborg, E., Hammarstrand, L., Pollefeys, M., Sattler, T., & Kahl, F. (2019). A cross-season correspondence dataset for robust semantic segmentation. In *CVPR* (pp. 9532–9542).

Lategahn, H., Beck, J., Kitt, B., & Stiller, C. (2013). How to learn an illumination robust image feature for place recognition. In *2013 IEEE intelligent vehicles symposium (IV)* (pp. 285–291). IEEE.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *IJCV, Springer*, *60*(2), 91–110.

Lowry, S., Sünderhauf, N., Newman, P., Leonard, J. J., Cox, D., Corke, P., et al. (2015). Visual place recognition: A survey. *IEEE Transactions on Robotics*, *32*(1), 1–19.

Maddern, W., Milford, M., & Wyeth, G. (2012). CAT-SLAM: Probabilistic localisation and mapping using a continuous appearance-based trajectory. *IJRR*, *31*(4), 429–451.

Maddern, W., Pascoe, G., Linegar, C., & Newman, P. (2017). 1 year, 1000 km: The oxford robotcar dataset. *The International Journal of Robotics Research*, *36*(1), 3–15.

Masone, C., & Caputo, B. (2021). A survey on deep visual place recognition. *IEEE Access*, *9*, 19516–19547.

McManus, C., Upcroft, B., & Newmann, P. (2014). Scene signatures: Localised and point-less features for localisation. In *Robotics, science and systems conference*.

Mei, C., Sibley, G., Cummins, M., Newman, P., & Reid, I. (2009). A constant-time efficient stereo slam system. In *Proceedings of the British machine vision conference* (Vol. 1). BMVA Press

Merrill, N., & Huang, G. (2018). Lightweight unsupervised deep loop closure. *Robotics Science and Systems Conference*. arXiv preprint arXiv:1805.07703.

Milford, M. (2013). Vision-based place recognition: How low can you go? *The International Journal of Robotics Research*, *32*(7), 766–789.

Milford, M. J., & Wyeth, G. F. (2012). SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *International conference on robotics and automation* (pp. 1643–1649). IEEE.

Mishkin, D., Perdoch, M., & Matas, J. (2015). Place recognition with WxBS retrieval. In *PCVPR 2015 workshop on visual place recognition in changing environments* (Vol. 30).

Mohan, A., Bailey, R., Waite, J., Tumblin, J., Grimm, C., & Bodenheimer, B. (2007). Tabletop computed lighting for practical digital photography. *IEEE Transactions on Visualization and Computer Graphics*, *13*(4), 652–662.

Mount, J., & Milford, M. (2016). 2d visual place recognition for domestic service robots at night. In *2016 IEEE international conference on robotics and automation (ICRA)* (pp. 4822–4829). IEEE.

Mousavian, A., Košecká, J., & Lien, J. M. (2015). Semantically guided location recognition for outdoors scenes. In *2015 IEEE international conference on robotics and automation (ICRA)* (pp. 4882–4889). IEEE.

Murillo, A. C., & Kosecka, J. (2009). Experiments in place recognition using gist panoramas. In *ICCV workshops* (pp 2196–2203). IEEE.

Murillo, A. C., Guerrero, J. J., & Sagues, C. (2007). Surf features for efficient robot localization with omnidirectional images. In *Proceedings of IEEE ICRA* (pp. 3901–3907).

Murmann, L., Davis, A., Kautz, J., & Durand, F. (2016). Computational bounce flash for indoor portraits. *ACM Transactions on Graphics (TOG)*, *35*(6), 1–9.

Murmann, L., Gharbi, M., Aittala, M., & Durand, F. (2019). A multi-illumination dataset of indoor object appearance. In *2019 IEEE international conference on computer vision (ICCV)*.

Nardi, L., Bodin, B., Zia, M. Z., Mawer, J., Nisbet, A., Kelly, P. H., Davison, A. J., Luján, M., O'Boyle, M. F., Riley, G., et al. (2015). Introducing slambench, a performance and accuracy benchmarking methodology for slam. In *2015 IEEE international conference on robotics and automation (ICRA)* (pp. 5783–5790). IEEE.

Naseer, T., Oliveira, G.L., Brox, T., & Burgard, W. (2017). Semantics-aware visual localization under challenging perceptual conditions. In *2017 IEEE ICRA* (pp. 2614–2620).

Noh, H., Araujo, A., Sim, J., Weyand, T., & Han, B. (2017). Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international conference on computer vision* (pp. 3456–3465).

Odo, A., McKenna, S., Flynn, D., & Vorstius, J. (2020). 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* (pp. 566–573). SciTePress.

Oliva, A., & Torralba, A. (2006). Building the gist of a scene: The role of global image features in recognition. *Progress in Brain Research*, *155*, 23–36.

Paul, R., Feldman, D., Rus, D., & Newman, P. (2014). Visual precis generation using coresets. In *2014 IEEE international conference on robotics and automation (ICRA)* (pp. 1304–1311). IEEE.

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

Pepperell, E., Corke, P. I., & Milford, M. J. (2015). Automatic image scaling for place recognition in changing environments. In *2015 IEEE international conference on robotics and automation (ICRA)* (pp. 1118–1124). IEEE.

Perronnin, F., Liu, Y., Sánchez, J., & Poirier, H. (2010). Large-scale image retrieval with compressed fisher vectors. In *2010 IEEE computer society conference on computer vision and pattern recognition* (pp. 3384–3391). IEEE.

Philbin, J., Chum, O., Isard, M., Sivic, J., & Zisserman, A. (2007). Object retrieval with large vocabularies and fast spatial matching. In *IEEE conference on computer vision and pattern recognition*.

Philbin, J., Chum, O., Isard, M., Sivic, J., & Zisserman, A. (2008). Lost in quantization: Improving particular object retrieval in large scale image databases. In *IEEE conference on computer vision and pattern recognition*.

Porav, H., Maddern, W., & Newman, P. (2018). Adversarial training for adverse conditions: Robust metric localisation using appearance transfer. In *2018 IEEE international conference on robotics and automation (ICRA)* (pp. 1011–1018). IEEE.

Radenović, F., Iscen, A., Tolias, G., Avrithis, Y., & Chum, O. (2018). Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Radenović, F., Tolias, G., & Chum, O. (2018). Fine-tuning CNN image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(7), 1655–1668.

Ranganathan, A. (2013). Detecting and labeling places using runtime change-point detection and place labeling classifiers. US Patent 8,559,717.

Revaud, J., Almazán, J., Rezende, R. S., & Souza, C. R. D. (2019a). Learning with average precision: Training image retrieval with a listwise loss. In *Proceedings of the IEEE international conference on computer vision* (pp. 5107–5116).

Revaud, J., De Souza, C., Humenberger, M., & Weinzaepfel, P. (2019b). R2d2: Reliable and repeatable detector and descriptor. In *Advances in neural information processing systems* (pp. 12405–12415).

Robertson, D. P., & Cipolla, R. (2004). An image-based system for urban navigation. In *BMVC* (Vol. 19, p. 165). Citeseer.

Ros, G., Sellart, L., Materzynska, J., Vazquez, D., & Lopez, A. M. (2016). 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* (pp. 3234–3243).

Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. In *ECCV* (pp. 430–443). Springer.

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

Sarlin, P. E., Cadena, C., Siegwart, R., & Dymczyk, M. (2019). From coarse to fine: Robust hierarchical localization at large scale. In *CVPR* (pp .12716–12725).

Sattler, T., Havlena, M., Schindler, K., & Pollefeys, M. (2016). Large-scale location recognition and the geometric burstiness problem. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1582–1590).

Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., Safari, D., Okutomi, M., Pollefeys, M., Sivic, J., et al. (2018). Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8601–8610).

Schönberger, J. L., Pollefeys, M., Geiger, A., & Sattler, T. (2018). Semantic visual localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6896–6906).

Se, S., Lowe, D., & Little, J. (2002). Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *IJRR*, *21*(8), 735–758.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2014). Overfeat: Integrated recognition, localization and detection using convolutional networks. In *2nd International conference on learning representations, ICLR 2014*.

Siméoni, O., Avrithis, Y., & Chum, O. (2019). Local features and visual words emerge in activations. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 11651–11660).

Singh, G., & Kosecka, J. (2010). Visual loop closing using gist descriptors in manhattan world. In *ICRA omnidirectional vision workshop* (pp. 4042–4047).

Sivic, J., & Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *Null* (p. 1470). IEEE.

Skinner, J., Garg, S., Sünderhauf, N., Corke, P., Upcroft, B., & Milford, M. (2016). High-fidelity simulation for evaluating robotic vision performance. In *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 2737–2744). IEEE.

Skrede, S. (2013). Nordland dataset. https://bit.ly/2QVBOym.

Stenborg, E., Toft, C., & Hammarstrand, L. (2018). Long-term visual localization using semantically segmented images. In *2018 IEEE ICRA* (pp. 6484–6490).

Stumm, E., Mei, C., & Lacroix, S. (2013). Probabilistic place recognition with covisibility maps. In *IROS* (pp. 4158–4163). IEEE.

Sturm, J., Engelhard, N., Endres, F., Burgard, W., & Cremers, D. (2012). A benchmark for the evaluation of RGB-D slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*. (pp. 573–580). IEEE.

Sünderhauf, N., & Protzel, P. (2011). Brief-gist-closing the loop by simple means. In *IROS* (pp. 1234–1241). IEEE.

Sünderhauf, N., Neubert, P., & Protzel, P. (2013). 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)* (p. 2013). Citeseer.

Sünderhauf, N., Shirazi, S., Dayoub, F., Upcroft, B., & Milford, M. (2015). On the performance of convnet features for place recognition. In *IROS* (pp. 4297–4304). IEEE.

Talbot, B., Garg, S., & Milford, M. (2018). OpenSeqSLAM2. 0: An open source toolbox for visual place recognition under changing conditions. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)* (pp. 7758–7765). IEEE.

Tipaldi, G. D., Spinello, L., & Burgard, W. (2013). Geometrical flirt phrases for large scale place recognition in 2d range data. In *2013 IEEE international conference on robotics and automation* (pp. 2693–2698). IEEE.

Tolias, G., Avrithis, Y., & Jégou, H. (2013). To aggregate or not to aggregate: Selective match kernels for image search. In *Proceedings of the IEEE international conference on computer vision* (pp. 1401–1408).

Tolias, G., Avrithis, Y., & Jégou, H. (2016a). Image search with selective match kernels: aggregation across single and multiple images. *International Journal of Computer Vision*, *116*(3), 247–261.

Tolias, G., Sicre, R., & Jégou, H. (2016b). Particular object retrieval with integral max-pooling of CNN activations. In *ICLR*. arXiv:1511.05879.

Tomită, M. A., Zaffar, M., Milford, M., McDonald-Maier, K., & Ehsan, S. (2020). ConvSequential-SLAM: A sequence-based, training-less visual place recognition technique for changing environments. arXiv preprint arXiv:2009.13454.

Tomită, M. A., Zaffar, M., Milford, M., McDonald-Maier, K., & Ehsan, S. (2021). Sequence-based filtering for visual route-based navigation: Analysing the benefits, trade-offs and design choices. arXiv preprint arXiv:2103.01994.

Topp, E. A., & Christensen, H. I. (2008). Detecting structural ambiguities and transitions during a guided tour. In *2008 IEEE international conference on robotics and automation* (pp. 2564–2570). IEEE.

Torii, A., Arandjelovic, R., Sivic, J., Okutomi, M., Pajdla, T. (2015). 24/7 Place recognition by view synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1808–1817).

Torii, A., Sivic, J., Pajdla, T., & Okutomi, M. (2013). Visual place recognition with repetitive structures. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 883–890).

Torii, A., Taira, H., Sivic, J., Pollefeys, M., Okutomi, M., Pajdla, T., & Sattler, T. (2019). Are large-scale 3d models really necessary for accurate visual localization? *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

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

Wang, J., Zha, H., & Cipolla, R. (2005). Combining interest points and edges for content-based image retrieval. In *IEEE international conference on image processing 2005* (Vol. 3, pp. III–1256). IEEE.

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

Weyand, T., Araujo, A., Cao, B., & Sim, J. (2020). 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* (pp. 2575–2584).

Ye, Y., Cieslewski, T., Loquercio, A., & Scaramuzza, D. (2017). Place recognition in semi-dense maps: Geometric and learning-based approaches. In *British machine vision conference (BMVC)*.

Yi, K. M., Trulls, E., Lepetit, V., & Fua, P. (2016). Lift: Learned invariant feature transform. In *European conference on computer vision*. (pp 467–483). Springer.

Zaffar, M., Ehsan, S., Milford, M., & Maier, K. M. (2018). Memorable maps: A framework for re-defining places in visual place recognition. arXiv preprint arXiv:1811.03529.

Zaffar, M., Ehsan, S., Milford, M., & McDonald-Maier, K. (2020). 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.

Zaffar, M., Khaliq, A., Ehsan, S., Milford, M., Alexis, K., & McDonald-Maier, K. (2019a). Are state-of-the-art visual place recognition techniques any good for aerial robotics? In *ICRA 2019 workshop on aerial robotics*. arXiv preprint arXiv:1904.07967.

Zaffar, M., Khaliq, A., Ehsan, S., Milford, M., & McDonald-Maier, K. (2019b). Levelling the playing field: A comprehensive comparison of visual place recognition approaches under changing conditions. In *IEEE ICRA workshop on database generation and benchmarking*. arXiv preprint arXiv:1903.09107.

Zeng, F., Jacobson, A., Smith, D., Boswell, N., Peynot, T., & Milford, M. (2019). Lookup: Vision-only real-time precise underground localisation for autonomous mining vehicles. In *2019 International conference on robotics and automation (ICRA)* (pp. 1444–1450). IEEE.

Zhang, X., Wang, L., & Su, Y. (2021). Visual place recognition: A survey from deep learning perspective. *Pattern Recognition*, *113*, 107760.