POLITECNICO DI TORINO

Master's Degree in Data Science and Engineering

Master's Degree Thesis

Optimizing Image Retrieval for Robust Visual Localization

Supervisors

Prof. Carlo MASONE

Dr. Gabriele TRIVIGNO

Dr. Gabriele Moreno BERTON

Candidate

Pablo TORASSO

February 2025

Summary

Visual localization, the process of determining a camera's exact 6-DoF pose within a known environment, is fundamental to applications such as autonomous vehicles, augmented reality, and robotics. Traditional methods like GNSS offer only coarse position estimates and are often unreliable in indoor or visually challenging scenarios, underscoring the need for alternative, high-precision approaches. This thesis addresses these challenges by focusing on the image retrieval component within the visual localization pipeline, a critical stage that narrows the search space for computationally expensive local feature matching and thereby enhances overall system efficiency and accuracy.

The research systematically evaluates state-of-the-art image retrieval models, including NetVLAD, AP-GeM, and SALAD, under diverse environmental conditions such as varying illumination, seasonal changes, and significant viewpoint differences. A key aspect of this study is the analysis of image angular diversity and its impact on retrieval performance, which reveals the sensitivity of current methods to changes in camera orientation. Additionally, the integration of local feature descriptors with global image representations is explored to further improve discrimination between similar scenes and reduce false positives.

Through comprehensive experiments and performance benchmarks, the thesis develops optimization strategies that enhance performances of visual localization systems. The results provide practical guidelines for deploying advanced image retrieval techniques in large-scale, real-world environments, thereby advancing the state-of-the-art in visual localization technology.

Acknowledgements

Desidero esprimere la mia più profonda gratitudine alla mia famiglia, a Federica e a mio figlio, che con il loro affetto e supporto mi hanno aiutato a mantenere alto il morale nei momenti di stress e concitazione durante questo percorso di laurea magistrale. Senza di loro, affrontare le difficoltà sarebbe stato molto più arduo.

Un ringraziamento speciale va ai miei genitori, che con il loro esempio e i valori che mi hanno trasmesso hanno posto le basi su cui ho potuto costruire il mio percorso. È grazie a loro se oggi ho avuto la possibilità di raggiungere questo importante traguardo.

Ringrazio di cuore gli amici e i compagni di viaggio che hanno condiviso con me questa avventura. Il loro sostegno, la loro compagnia e i momenti vissuti insieme hanno reso questo percorso non solo più leggero, ma anche più ricco e significativo.

Infine, un sincero ringraziamento ai miei relatori e correlatori, che con la loro guida e i loro preziosi consigli mi hanno accompagnato nella realizzazione di questo progetto di tesi. Il loro supporto è stato fondamentale per portare a termine questo lavoro con impegno e dedizione.

Grazie di cuore a tutti.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background and Motivation

Visual localization, the process of determining the exact 6-DoF pose of a camera within a known environment, is a fundamental problem in computer vision with applications across multiple domains. Its applications span diverse fields, from autonomous vehicles and augmented reality (AR) to robotics, navigation systems. Precise visual localization enables robust positioning and orientation estimation, which is crucial for tasks such as self-driving vehicles, AR-enhanced experiences, and simultaneous localization and mapping (SLAM) in robotics. For instance, in autonomous vehicles, Visual localization ensures that the vehicle can accurately interpret its environment and determine its position in relation to the map, enabling safe and efficient navigation. Similarly, in AR applications, precise localization enhances the user's experience by anchoring virtual objects to real-world coordinates, creating seamless interactions between the digital and physical worlds.

Many modern technologies, such as self-driving cars, mobile robots, and AR applications, interact with their environment in different ways, making accurate location determination essential. The most widely known approach for obtaining location information is through global navigation satellite systems (GNSS), such as GPS. However, GNSS-based methods come with limitations, they are ineffective indoors, provide location accuracy within only a few meters, and lack orientation data. While combining GPS with a compass can offer an estimated position accurate to 1/2 meters and an orientation within 10 degrees, applications like robotic navigation and self-driving vehicles demand much higher precision. Furthermore, since satellite-based methods are unreliable indoors, alternative geolocalization techniques must be explored to address these challenges.

Despite its significance, visual localization presents numerous challenges. Environmental variations, including changes in lighting, weather, and seasonal conditions, complicate scene recognition and localization. Furthermore, issues such as occlusions, repetitive structures, and large viewpoint changes add additional complexity. Addressing these challenges requires sophisticated methods that can reliably analyze and interpret visual data while maintaining computational efficiency and scalability.

One of the most effective approaches to visual localization is structure-based localization (SBL). This method relies on matching local keypoints from a query image to a pre-constructed 3D global model. By triangulating these keypoints, it estimates the precise pose of the camera. Structure-based localization has demonstrated outstanding performance in many scenarios due to its ability to leverage detailed geometric information. However, it also faces limitations, particularly in large-scale environments, where computational costs and memory requirements can become prohibitive. Additionally, SBL may fail if:

1. Fewer than two retrieved database images contain the same place as the query image.
2. The viewpoint difference between the retrieved images and/or between the query and retrieved images is too large.
3. The baseline between retrieved database images is too small for stable triangulation.

Thus, effective visual localization requires retrieving a diverse set of images depicting the same scene from multiple viewpoints. This necessity highlights the importance of image representations that are robust but not entirely invariant to viewpoint changes.

Within the Visual localization pipeline, image retrieval plays a pivotal role in overcoming these challenges and ensuring the pipeline's efficiency and scalability. Image retrieval is the process of identifying and ranking the most visually similar images from a large database of geotagged reference images, given a query image. By narrowing down the search space, it enables the pipeline to focus computational resources on the most relevant candidates, facilitating accurate and efficient localization.

Image retrieval is employed at two critical stages of the visual localization pipeline:

1. **During Database Pair Matching:** Before constructing the 3D map, image retrieval identifies pairs of images in the database that are likely to correspond to spatially close locations. This reduces the number of image pairs requiring computationally expensive local feature matching, making the map-building process feasible for large datasets.

2. **During Query Localization:** When a query image is presented, image retrieval is used to identify the top-N most relevant database images, effectively reducing the search space for local feature matching. This second application ensures that the pipeline can scale efficiently to handle large databases while maintaining high localization accuracy.

The performance of the image retrieval system directly impacts the effectiveness of the entire localization pipeline. A robust retrieval mechanism enhances accuracy by identifying the most relevant images while minimizing false positives. Additionally, it improves computational efficiency by significantly reducing the number of candidate images for subsequent processing stages. As highlighted in prior research [1, 2], optimizing the image retrieval component is crucial for addressing real-world challenges in visual localization, and the development of specialized feature representations tailored to this task presents a promising research direction.

Image retrieval algorithms are designed to fetch the most relevant images from extensive databases based on a given query, yet their requirements vary depending on the application. For example, in generic visual search tasks, retrieval focuses on identifying all images depicting the same content as the query image, irrespective of variations in viewpoint or camera angle. However, in the context of visual localization, robustness to viewpoint changes is not always desirable. Benchmarking studies on image retrieval for localization have emphasized the need for retrieval strategies that cater specifically to localization requirements. While most modern localization pipelines rely on deep-learned image descriptors optimized to retrieve images of the same landmark as the query, they often overlook camera pose information. Since pose approximation benefits from retrieving database images captured from similar viewpoints as the query, tailoring retrieval methods to incorporate camera pose considerations could further enhance localization accuracy [1].

Refining the image retrieval process to better align with the specific needs of visual localization, can improve both the efficiency and precision of these systems. This underscores the importance of continued research into specialized retrieval approaches that bridge the gap between traditional image retrieval and pose-aware localization techniques.

1.2 Objectives of the Thesis

The primary aim of this thesis is to enhance the efficiency and accuracy of the visual localization pipeline that utilizes structure-based approach by focusing on the image retrieval component only in the query localization task. To achieve this, the research is structured around the following key objectives:

1. **Evaluate the Impact of Image Angular Diversity:** This objective focuses on understanding how variations in the angular orientation of images affect retrieval performance. By systematically analyzing images captured from different angles, the thesis seeks to quantify the influence of viewpoint differences on the ability of state-of-the-art models to correctly identify geospatially relevant images. This evaluation will provide insights into the robustness of current methods against variations in image angles and help identify potential areas for improvement.

2. **Test and Compare Advanced Image Retrieval Models:** A critical component of this research involves a comprehensive assessment of advanced image retrieval models, including NetVLAD, AP-GeM, and SALAD. The goal is to benchmark these models under consistent experimental settings, comparing their performance in terms of accuracy, by measuring the precision of retrieval results in diverse environmental conditions and across varying scales.

3. **Explore the Integration of Local Feature Descriptors:** While global image representations provide a solid foundation for image retrieval, integrating local features can offer finer granularity and improved discrimination between similar scenes. This objective aims to:

    - Investigate how local feature descriptors can complement global representations.
    - Develop strategies for effectively combining these features to reduce false positives and enhance overall retrieval accuracy.
    - Assess the impact of this integration on the visual localization pipeline, particularly in scenarios with high visual variability.

Through these objectives, the thesis strives to advance the state-of-the-art in visual localization by not only improving the core image retrieval process but also by ensuring that these improvements translate into more reliable and efficient localization in practical scenarios.

Chapter 2

Related Work

Visual localization offers the potential to achieve highly accurate indoor and outdoor positioning, often reaching accuracies on the order of a few centimeters and degrees. For instance, in self-driving cars, the rich visual data captured by onboard cameras enables the establishment of correspondences between a query image and a pre-constructed world representation, or "visual map." This visual map may take the form of a 3D reconstruction, a collection of geotagged images, or even a deep neural network, thereby providing the basis for computing the camera's precise position relative to the environment. Indoors, such localization techniques are invaluable not only for robot navigation and augmented reality (AR) applications but also for autonomous vehicles operating in environments where GNSS signals are unreliable, such as tunnels or underground parking structures.

Figure 2.1: Example of typical challenges visual localization is facing. The images depict the same building under varying illumination, different scales, seasonal changes, and occlusions. Images from [3].

Despite its promise, visual localization must overcome several significant challenges, including:

- Illumination Changes: Can a system localize a nighttime image when the reference images were captured during the day?
- Dynamic Scenes: How does one account for moving objects that may not have

been present during the creation of the map?

- Seasonal and Weather Variations: Variations in appearance due to different seasons or weather conditions can greatly affect matching.
- Occlusions: The presence of objects or people that partially obscure the scene can complicate localization.
- Viewpoint Changes: Strong differences in the viewpoints between the reference and query images further complicate the task.

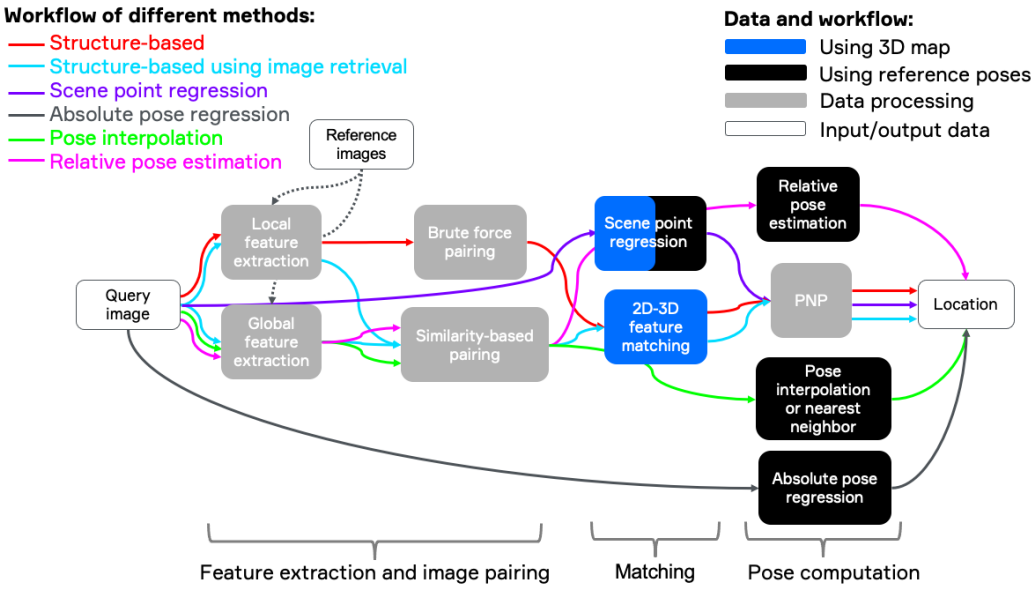


Figure 2.2: Overview of different methods for visual localization: all visual localization approaches need a set of pose-tagged reference images to create a map or some other representation of the environment. The methods then differ in their workflow: Structure-based methods use local feature correspondences to generate a 3D map and to localize an image within the map. Image retrieval can be used to reduce the search space by pairing only similar images instead of all possibilities. Alternatively, these similar images can be used for pose interpolation or relative pose estimation. Scene point regression methods can directly determine correspondences between 2D pixel locations and 3D points using a deep neural network (DNN) and compute the camera pose similar to structure-based methods. Finally, absolute pose regression methods estimate the pose end-to-end using a DNN. Figure and description from [3].

To address these challenges, several benchmark datasets and evaluation challenges have been developed. These datasets are designed to test visual localization pipelines under conditions with significant appearance variations, including those

induced by seasonal changes (summer, winter, spring, etc.) and differing illumination (dawn, dusk, day, night). In parallel, the growing interest in visual localization has spurred the production of a wealth of online resources which provide valuable insights into state-of-the-art techniques.

Many strategies have been developed over the years to tackle these issues. As shown in figure 2.2, a range of prominent methods has been explored. Traditional, structure-based techniques rely on constructing a 3D representation, typically in the form of point clouds, of the environment and then matching local features between the query image and this 3D map.

To streamline this process, image retrieval methods are often employed, limiting the search to only the most visually similar reference images rather than considering every possibility. Alternatively, these similar images can be used either to directly interpolate the camera pose or to estimate the relative pose between the query and the retrieved references without building a complete 3D model. Additionally, scene point regression techniques employ a deep neural network (DNN) to establish correspondences between 2D pixel locations and 3D points, ultimately computing the camera pose in a manner analogous to structure-based approaches.

Approach	**3D Map**	**Pros**	**Cons**
Structure-based	yes	Perform very well in most scenarios	Challenging in large environments in terms of processing time and memory consumption
Structure-based with image retrieval	yes	Improve speed and robustness for large-scale settings	Quality heavily relies on image retrieval
Scene point regression	yes/no	Very accurate position in small-scale settings	To be improved in large environments
Absolute pose regression	no	Fast pose approximation, can be trained for certain challenges	Low accuracy
Pose interpolation	no	Fast and lightweight	Quality relies heavily on image retrieval and only provides a rough pose
Relative pose estimation	no	Fast and lightweight	Quality relies heavily on image retrieval and, e.g., local feature matches or a DNN used for relative pose estimation

Table 2.1: Comparison of different visual localization methods.

Modern scene point regression methods typically make use of a 3D reconstruction during the training phase, even though they do not depend on it during inference. In contrast, absolute pose regression methods employ a DNN to estimate the pose in an end-to-end fashion. As summarized in table 2.1, these approaches differ in their ability to generalize and in localization accuracy. Some methods require detailed 3D reconstructions, while others operate solely with pose-tagged reference images. Although 3D reconstructions can yield highly accurate poses, they often entail substantial computational and storage demands, particularly in large or dynamic environments where updates are frequent.

2.1 Structure-based Methods

Structure-based methods remain a cornerstone in visual localization research. The conventional approach begins by constructing a 3D map, a process that gives rise to the term "structure", and subsequently localizing the query image within that map [4, 5, 6, 7, 8, 9]. This process, known as structure from motion (SfM) [5, 9], involves estimating camera poses and reconstructing 3D points from a collection of images taken from multiple viewpoints. Modern SfM pipelines not only optimize the camera poses but also refine the 3D point positions to improve overall map consistency and robustness. Pixel-level correspondences across images are generated automatically using local features [10, 11]. These correspondences may be established by exhaustively comparing every image pair (brute force) or by selecting pairs based on image similarity (as discussed in the next section).

A local feature is defined by an exact pixel location (the keypoint) and a distinctive descriptor, often derived from the pixel's surrounding neighborhood. Early approaches relied on handcrafted feature extractors, such as the popular SIFT descriptor [12]. However, these handcrafted methods can struggle with challenges like drastic illumination changes or seasonal variations. In response, several data-driven methods have recently emerged [11, 13, 14, 15], including end-to-end deep architectures that learn local features, as detailed in [10, 16]. Such advancements have improved both the robustness and discriminative power of the descriptors.

Since the 3D map is built using image descriptors, these same descriptors facilitate the establishment of 2D-3D correspondences between the query image's keypoints and the 3D points in the map. Once these matches are secured, the camera pose is computed using perspective-n-point (PnP) solvers [17]. To handle potential outliers among the matches, PnP is typically solved within a RANSAC loop [18, 19], with recent improvements incorporating guided sampling and adaptive thresholds for enhanced performance under challenging conditions.

Mapping extensive areas can lead to massive 3D models. For example, the

Aachen-Day-Night dataset [20] comprises between 700k and 2.5M 3D points, depending on the number of image pairs used, despite covering only the historic inner city of Aachen, Germany. In such scenarios, matching every query keypoint with all 3D points becomes computationally infeasible. To address this, image retrieval methods (discussed below) are applied to first select the most relevant images, effectively confining the keypoint matching to a localized region [21, 22]. This strategy not only reduces computational complexity but also improves the accuracy of subsequent pose estimation by focusing on spatially coherent areas.

In some cases, rather than relying on a global map, the retrieved images can be used to assemble a temporary local map. While the localization process remains similar, the drawback is that if too few images depicting the same scene are retrieved, the resulting local map may be insufficiently robust. Additionally, dynamic changes in the environment may render portions of a global map outdated, making local mapping an attractive alternative in rapidly evolving settings.

Another approach within structure-based methods is scene point regression, wherein 2D-3D correspondences are directly predicted using a DNN [23, 24, 25, 26] or a random forest [27]. These techniques are particularly beneficial in scenarios where traditional feature matching is challenged by significant viewpoint variations or textureless regions, though their success is highly dependent on the diversity and quality of the training data.

2.2 Image Retrieval-based Methods

Image retrieval plays a crucial role in visual localization by supporting structure-based techniques in large-scale environments, such as shopping centers, airports, and urban areas, and by serving as an independent alternative when a detailed 3D structure is unavailable. Typically, database images are stored with either approximate location information (such as GPS in geo-localization scenarios) or exact 6 degrees of freedom (DoF) poses (in visual localization). This dual storage approach offers flexibility, with the choice often dictated by the precision of available sensor data.

The primary goal is to identify a subset of images that closely resemble the query image based on a chosen representation. This initial selection is often refined through a re-ranking process using techniques like query expansion or filtering. In the context of visual localization, a retrieved image is considered relevant if it captures the same scene (e.g., a landmark) as the query or if it was taken in close geographic proximity. When the criterion is scene similarity, the process is known as “landmark retrieval”; when it is based on proximity, it is referred to as “geo-localization” or “place recognition”. Consequently, the architectures for landmark retrieval and place recognition share many similarities, particularly among recent

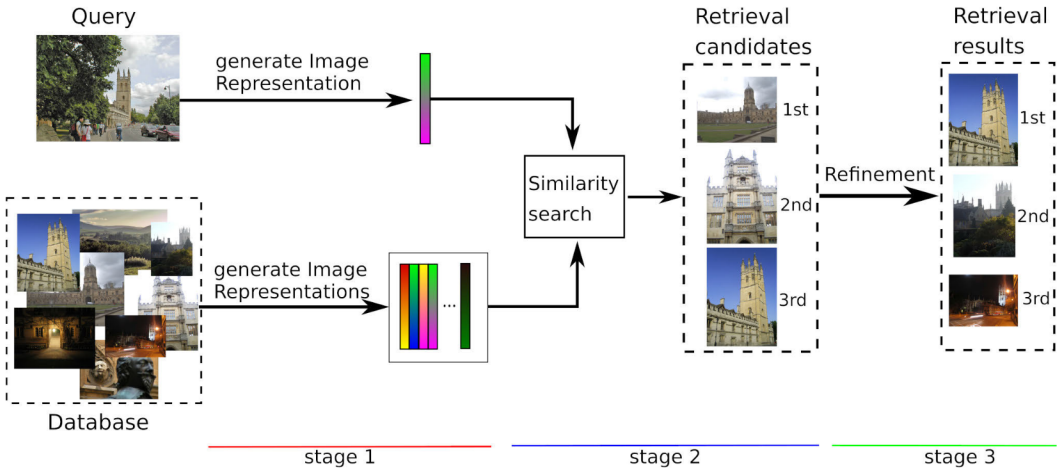


Figure 2.3: The conceptual architecture of a Visual Geolocation (VG) system formulated as an Image Retrieval task. Initially, the system extracts feature representations from both the query and database images, using either handcrafted or deep-learning-based methods. Next, a similarity search is conducted to compare the extracted features, generating a ranked list of potential matches. Optionally, a post-processing step further refines the final retrieval results for improved accuracy. Figure from [28].

deep learning-based models [29, 30]. Additionally, the adoption of convolutional neural networks for feature extraction has significantly enhanced retrieval accuracy by capturing more abstract, semantically meaningful representations.

Historically, many image retrieval systems utilized variations of the "bag of visual words" model. More recently, however, features extracted from DNN activations, which capture high-level semantic information, have proven to be highly effective. The performance of these systems is further boosted when the networks are specifically trained for retrieval using ranking losses [31, 32, 33]. Moreover, recent research incorporating attention mechanisms has demonstrated improved robustness by dynamically weighting feature importance in complex scenes.

Image retrieval not only narrows the search space during the localization process when a 3D map is available, but it also offers a direct, albeit less precise, means of localization when no such map exists. For example, one can simply assign the pose of the nearest neighbor or interpolate among the top k retrieved poses [34, 35]. These approaches are particularly useful in applications requiring rapid, on-the-fly localization, with the initial estimates later refined through additional processing.

Alternatively, if the intrinsic parameters of the camera are known, the relative pose between the query image and a retrieved image can be estimated using local feature matches. With the absolute poses of the reference images available, the query's pose can then be deduced from these relative measurements [36, 37]. This method effectively bridges the gap between purely appearance-based and geometric

techniques, leveraging both types of information for improved accuracy.

A comprehensive benchmark evaluating image retrieval for visual localization is presented in [1]. The study compares four popular image representations across three tasks, using a global 3D map, constructing an on-the-fly local 3D map, and interpolating camera poses, and examines the relationship between landmark retrieval and visual localization. The findings underscore the need for retrieval methods that are robust to environmental variations and adaptable to the diverse requirements of different localization scenarios.

2.3 Pose Regression-based Methods

Recent breakthroughs in image classification, semantic segmentation, and image retrieval have paved the way for a deep CNN-based approach to visual localization. In this strategy, a CNN processes an RGB image and learns to directly regress the camera pose in an end-to-end manner. The core idea is that the low-level features developed for general vision tasks also encode valuable information for pose estimation, which can be leveraged through transfer learning from pre-trained CNNs. This approach has attracted significant interest due to its simplicity and potential for real-time deployment, although achieving high precision remains a challenge.

For instance, PoseNet [38] adapts well-known image classification architectures, such as VGGNet or ResNet, by replacing the final softmax layers with fully connected layers that output the 3D position and orientation of the camera, as depicted in figure 2.4. Despite its innovative design, PoseNet's performance highlights the inherent trade-offs in end-to-end learning, where the network must balance between general feature extraction and precise pose regression.

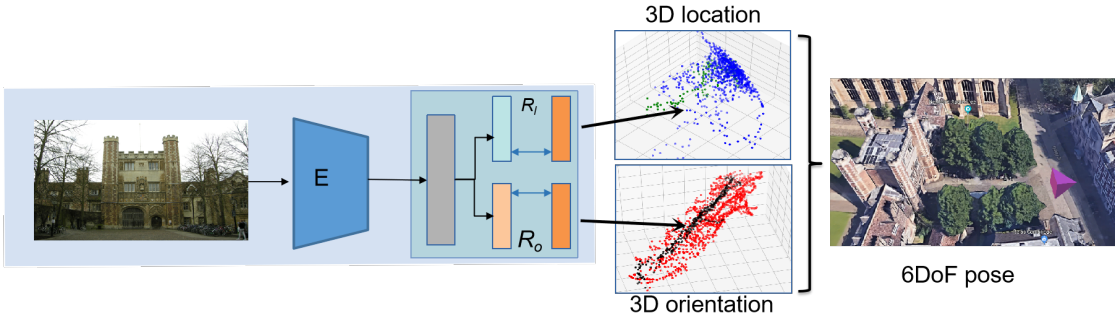


Figure 2.4: PoseNet architecture for absolute pose regression. Image encoder E, position regressor R_p and orientation regressor R_o produce 6DoF pose of the image. Figure and description from [3]. Image taken from the Cambridge Landmarks dataset [38].

End-to-end absolute pose regression offers several benefits. It eliminates the need

for manual feature engineering by relying on learned representations that are robust to changes in lighting and viewpoint. Additionally, these models typically require less memory and offer constant inference time compared to methods that depend on extensive 3D models. Transfer learning also facilitates effective training on moderately sized datasets. However, the accuracy of pose estimates from PoseNet, both in translation and rotation, is generally an order of magnitude lower than that achieved by state-of-the-art structure-based methods. This performance gap has spurred extensive research into refining network architectures and loss functions to better capture the complexities of 3D space.

Various enhancements to the PoseNet framework have been proposed, including the development of new loss functions [32], the incorporation of Long-Short-Term-Memory (LSTM) layers, and the integration of additional sensor data [39]. Moreover, VLocNet [40] introduced a joint framework for learning both absolute and relative poses, which was further refined in VLocNet++ [41] by incorporating semantic segmentation as an auxiliary task. These developments exemplify a growing trend toward multi-task learning, where complementary tasks provide additional context that significantly improves pose estimation.

Despite these advancements, fully end-to-end learning of the localization pipeline often results in suboptimal performance because it tightly couples the network to specific scene coordinates, essentially compressing an implicit map of the environment, which can hinder generalization. This tight coupling makes models particularly vulnerable to overfitting, especially when deployed in scenes that differ significantly from the training data.

To mitigate these limitations, newer hybrid methods have emerged that shift the focus toward localized sub-tasks while combining them with traditional image retrieval and structure-based techniques. For example, DSAC [42] utilizes geometric constraints to concentrate on tasks such as establishing 2D-3D correspondences. While this hybrid approach considerably improves pose accuracy, the resulting models tend to be scene-specific and may not generalize well to new environments. By integrating geometric priors, these models strike a balance between data-driven learning and model-based reasoning, offering a promising direction for future improvements.

Addressing the generalization challenge, the recent SANet [43] introduces a scene-agnostic neural framework for camera localization. By decoupling the model parameters from any particular scene, SANet leverages geometric cues from 3D point clouds, obtained via dense multi-view stereo (MVS) reconstructions from the top retrieved images, and jointly learns query-scene registration along with camera pose regression. This innovative approach represents a significant step toward developing robust localization systems that can adapt to a wide variety of environments without extensive retraining.

2.4 Benchmarking Image Retrieval for Visual Localization

The performance of image retrieval systems is pivotal in visual localization pipelines, as the quality of retrieval directly influences the efficiency and accuracy of subsequent pose estimation stages. In recent years, research has increasingly focused on the task of benchmarking retrieval methods within the context of visual localization, with an emphasis on understanding their individual strengths, limitations, and overall impact on localization performance. Two influential contributions in this area are *Investigating the Role of Image Retrieval for Visual Localization* [1] and *Benchmarking Image Retrieval for Visual Localization* [2].

The work from Pion et al. proposes a comprehensive benchmarking framework designed specifically for image retrieval in visual localization tasks. This framework extends conventional retrieval evaluation methods by incorporating localization-specific factors.

The authors present an evaluation protocol that unifies various retrieval metrics while integrating factors such as spatial consistency and geometric robustness. This approach allows for a more holistic assessment of retrieval methods by considering how well the retrieved images support accurate pose estimation.

The framework is tested across multiple challenging datasets, highlighting the trade-offs between computational efficiency and retrieval precision in different environmental settings. The analysis spans various scenarios, including urban, suburban, and dynamic environments, thereby demonstrating the scalability of the proposed metrics.

A systematic comparison is carried out between classical retrieval approaches and modern deep learning-based methods. The study reveals that while traditional methods may offer competitive performance in certain controlled scenarios, modern approaches, especially those employing deep networks, tend to perform better under complex, real-world conditions. Moreover, the paper provides detailed insights into how retrieval performance correlates with localization accuracy, thereby guiding the development of more effective retrieval systems.

By advocating for standardized benchmarking protocols, the study underlines the need for consistent evaluation practices in the community. This standardization not only aids in comparing results across different studies but also drives future research towards refining retrieval methodologies that can robustly support visual localization.

In a complementary effort, the subsequent work developed by the same authors provides an in-depth analysis of how the retrieval process affects overall localization accuracy. The authors evaluate various retrieval strategies under challenging conditions.

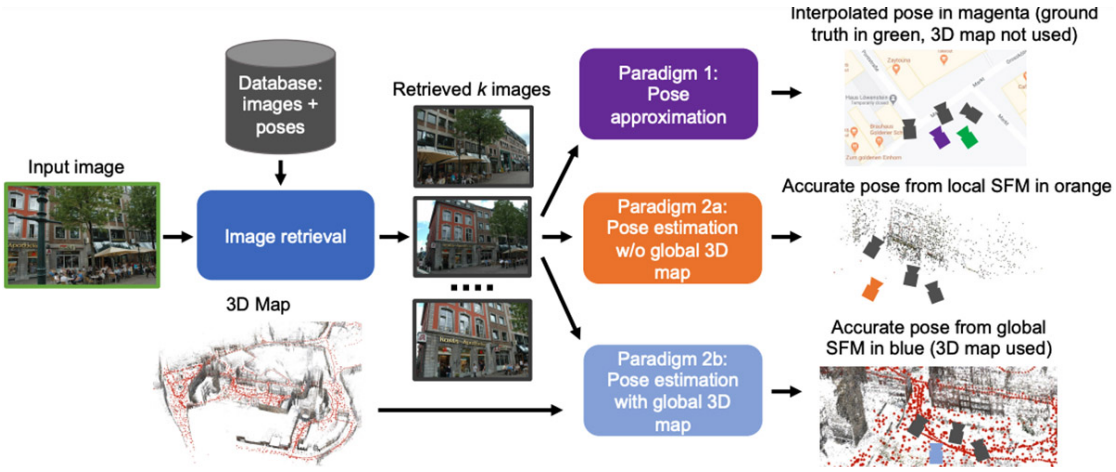


Figure 2.5: Pipeline used in [1, 2] to analyze the role of image retrieval in three visual localization paradigms through extensive experiments. Figure from [1, 2].

The paper introduces robust evaluation metrics that go beyond traditional precision and recall measures. These metrics are tailored to capture the nuances of visual localization, such as spatial consistency and the geometric alignment of retrieved images, which are crucial for accurate pose estimation.

By comparing both handcrafted features (e.g., SIFT [44]) and deep learning-based descriptors, the study illustrates the trade-offs inherent in different representation methods. The evaluation reveals that while deep representations (e.g., those derived from CNN activations) excel in capturing high-level semantic information, integrating them with local feature descriptors can further mitigate challenges in large-scale or dynamically changing environments.

A notable finding of this work is that even slight improvements in retrieval accuracy can lead to substantial gains in localization performance. The authors demonstrate that a more accurate retrieval stage reduces the number of candidate images for the subsequent matching process, thereby lowering computational costs and enhancing robustness against false positives. They also explore how environmental factors, such as lighting variations and occlusions, impact retrieval performance and, by extension, the overall localization accuracy.

The study lays the groundwork for standardized evaluation protocols by highlighting the importance of considering both global descriptor quality and the potential benefits of local feature integration. These protocols have since become a reference point for subsequent research in the field.

Both studies underscore the critical role that image retrieval plays in the visual localization pipeline. Their rigorous benchmarks reveal that:

- Enhanced retrieval methods not only reduce the computational burden during

the matching process but also improve the robustness of the subsequent pose estimation.

- Integrating local feature descriptors with global image representations can significantly reduce false positives, especially in environments characterized by high visual variability.
- The balance between retrieval accuracy and efficiency is crucial, particularly when scaling to large and complex datasets.

The findings from these benchmarking studies provide a roadmap for future research, highlighting that improvements in the retrieval component can lead to notable advancements in visual localization performance. They encourage the development of hybrid systems that effectively combine the strengths of both global and local feature extraction methodologies.

Building upon the insights and methodologies presented in these works, this thesis adopts their benchmarking principles and findings as a foundation for further exploration. By addressing the challenges identified in these studies and leveraging their suggested directions for future work, we aim to refine image retrieval techniques to enhance both accuracy and computational efficiency in visual localization. Specifically, this thesis extends their contributions by examining the impact of angular diversity in images retrieved with global features, exploring different global feature extractors, and developing novel approaches that integrate global and local descriptors while considering real-world deployment constraints. Ultimately, our goal is to push the boundaries of retrieval-based localization systems.

Chapter 3

Benchmarking Framework

In this chapter, we introduce a comprehensive benchmarking framework designed to evaluate state-of-the-art visual localization systems under real-world conditions. Visual localization is a crucial component in many applications, but its performance is highly dependent on sensor data quality and the effectiveness of feature extraction, matching, and mapping strategies. To assess these factors, the framework integrates multiple datasets and advanced retrieval models, rigorously measuring accuracy, robustness, and efficiency.

For our experiments, we adopted the visual localization pipeline proposed in the LaMAR paper [45]. This pipeline provides a strong baseline, offering a well-validated experimental setup for benchmarking different localization approaches. Given its structured design and prior validation in large-scale benchmarks, it serves as a reliable foundation for evaluating various feature extraction and matching strategies. The chapter begins with a detailed discussion of this pipeline and the central role of image retrieval in the localization process.

Next, we introduce the global feature extractors used in our benchmarking (Section 3.1). We evaluate three different extractors: NetVLAD and AP-GeM, both widely used in visual localization for their strong retrieval performance, and SALAD, a lightweight alternative prioritizing efficiency. These global descriptors were integrated into a fusion extractor, enabling us to analyze the impact of different retrieval strategies on localization accuracy. All three extractors were used off-the-shelf without additional fine-tuning.

Following this, in Section 3.2, we describe our approach to local feature extraction and matching. We employed SuperPoint as the keypoint detector and LightGlue for feature matching. SuperPoint provides robust and repeatable keypoints, while LightGlue efficiently matches local features, making it a strong choice for refining localization accuracy. This combination allows us to leverage both global and local features effectively, balancing retrieval efficiency with precise geometric alignment.

Then, in 3.3, we introduce the two datasets used in our evaluation: the LaMAR

dataset [45] and the Visual Benchmark in Rome (VBR) dataset [46]. We discuss their characteristics, differences, and how they were utilized in our research.

By using this setup, we aim to evaluate the trade-offs between retrieval precision, efficiency, and overall localization accuracy while maintaining comparability with existing benchmarks. The use of off-the-shelf global descriptors and robust local features ensures a structured and reproducible evaluation of different feature extraction methods.

3.0.1 Overview of the Visual Localization Pipeline

A visual localization pipeline is a systematic approach designed to estimate the geographic location of a query image by leveraging a combination of feature extraction, image matching, and spatial reasoning techniques. The pipeline integrates data from large-scale geotagged image databases and applies sophisticated algorithms to ensure accurate localization under varying conditions. This pipeline involves several key stages, each building upon the previous, to achieve efficient and precise localization. The pipeline we will consider here is the pipeline implemented and used by the paper *LaMAR: Benchmarking Localization and Mapping for Augmented Reality* [45], and it can be described as follows:

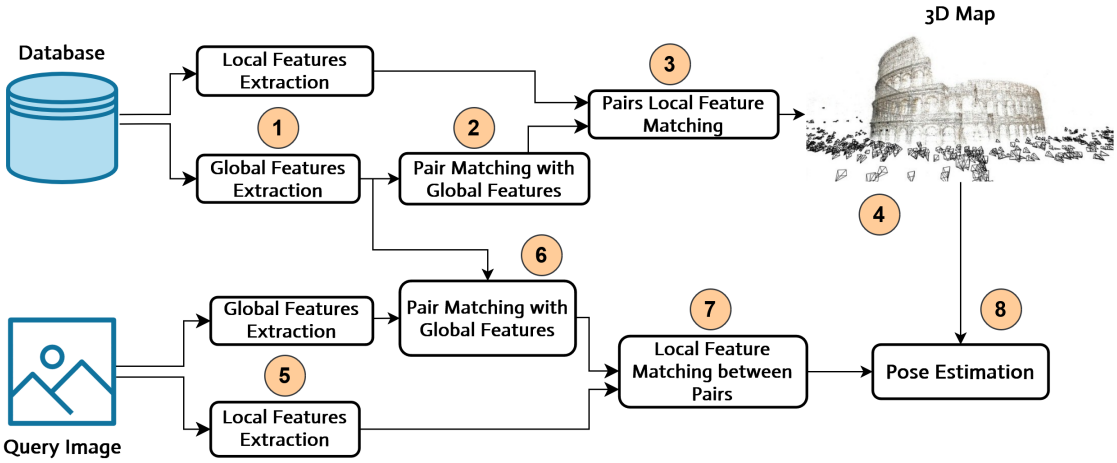


Figure 3.1: Lamar's visual localization pipeline. "3D Map" figure from[47].

1. **Extraction of Local and Global Features for the Database:**
   The pipeline begins with the preprocessing of the reference database of geotagged images. For each image in the database, both global and local features are extracted. Global features, such as those generated by deep learning models like NetVLAD or AP-GeM, provide compact, high-level representations of an image's overall visual content. Local features, such as SIFT [44] or ORB

[48] keypoints, capture detailed information about specific image regions. The combination of these features ensures the database is ready for robust and scalable matching.

2. **Matching Database Images Using Global Features (Pair Matching):** Image retrieval is used for the first time in the pipeline during this stage to perform pair matching within the database. By comparing the global features of images in the database, the system identifies pairs of images that are visually similar and likely to represent spatially proximate locations. This process significantly reduces the number of image pairs that need to undergo the computationally expensive local feature matching step. Without this retrieval-based filtering, the sheer number of image pairs in large databases would make subsequent steps infeasible.

3. **Local Feature Matching Between the Pairs:** After identifying candidate pairs of images using global features, local feature matching is performed to establish precise correspondences between keypoints in the paired images. These correspondences are essential for computing relative poses and identifying geometric relationships between images.

4. **Creation of the 3D Map:** Using the matched pairs and their local feature correspondences, a georeferenced 3D map of the environment is constructed. Structure-from-motion (SfM) techniques are employed to triangulate 3D points and estimate the relative poses of the cameras associated with the database images. The result is a dense 3D point cloud that encodes the spatial geometry of the scene, as well as the camera positions and orientations for all database images. This 3D map serves as the foundational reference for query localization.

5. **Extraction of Local and Global Features for the Queries:** When a query image is presented for localization, its local and global features are extracted using the same techniques applied to the database images. This ensures that the query features are compatible for comparison with the database features.

6. **Matching Queries and Database Images Using Global Features:** Image retrieval is used for the second time in the pipeline during this stage. The global features of the query image are matched against the global features of the database images to retrieve a ranked list of the top-N most relevant candidates. These candidates are the images in the database that are most likely to correspond to locations near the query. This retrieval step reduces the search space for the computationally intensive local feature matching, making the system more efficient.

7. **Local Feature Matching Between Query-Database Pairs:**
   For each candidate match identified in the previous step, local feature matching is performed to validate and refine the relationship between the query image and the database images. This step ensures that the retrieved images are geographically relevant and prepares the system for precise pose estimation.

8. **Pose Estimation of the Queries:**
   Using the keypoint correspondences between the query image and the 3D map, the pipeline estimates the exact pose of the query image. This involves determining the camera's position and orientation at the time the query image was captured. Perspective-n-Point (PnP) algorithms are often employed, leveraging the spatial relationships established during the map-building phase.

3.0.2 The Two-Phase Role of Image Retrieval

The image retrieval task plays a pivotal role in the pipeline, appearing twice to optimize the system's efficiency and scalability.

- **Building the 3D Map from the Database:**
  The creation of the 3D map is a critical step that relies heavily on the effective use of image retrieval to streamline the process. In the first use of image retrieval, global feature matching is performed within the database to identify pairs of images that are visually similar and likely to correspond to spatially adjacent locations. This step reduces the number of image pairs that need to undergo local feature matching, which is computationally expensive. By filtering out irrelevant or dissimilar images, the system focuses only on meaningful candidates, ensuring that computational resources are used efficiently.

  Once candidate pairs are identified, local feature matching is conducted to establish precise keypoint correspondences. These correspondences are then used to compute the relative poses of the database images, forming the basis for 3D map construction. Structure-from-motion (SfM) techniques are applied to generate a dense 3D point cloud that represents the spatial geometry of the scene, along with the georeferenced positions of the cameras. This map is essential for localization, as it provides the framework for pose estimation during the query phase.

- **During Query Localization:**
  Image retrieval is used again during the query phase to retrieve a ranked list of top-N database images that are most visually similar to the query image. This step narrows again the search space for local feature matching between the query and the database, ensuring that only the most relevant candidates

are considered. By minimizing unnecessary computations, this second usage of image retrieval improves the efficiency of the overall localization process.

Both phases highlight the importance of image retrieval as a tool for filtering and prioritizing candidate matches, allowing the pipeline to achieve both precision and computational efficiency. This two-phase application of image retrieval underscores its central role in balancing accuracy and scalability in modern visual localization systems.

3.1 Global Feature Extractors

In this section, we present an overview of the state-of-the-art models and techniques employed in our pipeline. The Lamar pipeline used in this work leverages a fusion of global feature extractors, namely, NetVLAD and AP-GeM, to obtain robust image representations. In addition, the SALAD model is utilized to provide an alternative retrieval baseline.

3.1.1 NetVLAD

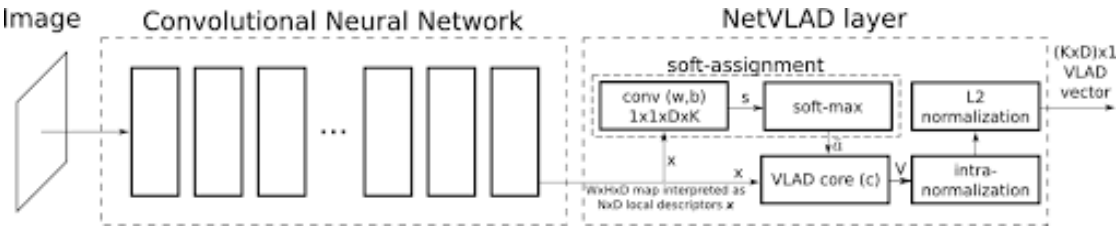


Figure 3.2: Netvlad architecture. Figure from [29]

NetVLAD [29] is a convolutional neural network architecture designed to generate compact global descriptors from input images, making it a critical component in image retrieval tasks. Building upon the traditional VLAD (Vector of Locally Aggregated Descriptors) method, NetVLAD incorporates a trainable clustering layer that aggregates local convolutional features into a single, fixed-dimensional global representation. This integration transforms the classic VLAD into a differentiable module, allowing the entire network to be trained end-to-end for the retrieval objective.

Historically, NetVLAD has had a significant impact on the field of visual localization. Introduced by Arandjelović et al. in 2016 [29], it marked a paradigm shift by effectively bridging the gap between hand-crafted feature aggregation techniques and deep learning. Prior to NetVLAD, many systems relied on engineered features like SIFT [44] combined with traditional VLAD, but these methods were

limited by their inability to leverage large amounts of training data in a unified framework. NetVLAD's innovative design not only provided improved robustness to variations in viewpoint, illumination, and scale, but also set a new standard for descriptor compactness and efficiency. Its success has spurred a wealth of research into learned image representations and has become a cornerstone in state-of-the-art place recognition and localization pipelines [49, 50, 51, 52, 53, 54, 55, 56].

One of the key innovations of NetVLAD is its ability to learn a set of cluster centers (or visual words) directly from data. Each local feature extracted from a convolutional layer is softly assigned to these learned centers, and the residuals between the features and the centers are aggregated. This process results in a descriptor that captures the distribution and arrangement of visual features in the image, effectively summarizing the overall scene context.

Given an image with a set of N local feature vectors $F = \{x_1, ..., x_N\}$, the global VLAD descriptor is a matrix v with dimension $K \times D$, where D is the dimensionality of the local features. Each element contains the sum of residuals of each local descriptor and the nearest word as:

$$V(j,k) = \sum_{i=1}^{N} a_k(x_i) \left(x_i(j) - c_k(j)\right) \tag{3.1}$$

To achieve trainability, the NetVLAD pooling must be differentiable. Thus, a soft assignment to codewords is adopted, and the descriptor is adjusted accordingly as:

$$v_{i,j} = \sum_{a=1}^{N} \frac{e^{w_i^T x_a - b_i}}{\sum_{i'} e^{w_{i'}^T x_a - b_{i'}}} (x_{a,j} - c_{i,j}) \tag{3.2}$$

in which each cluster has a set of independent trainable parameters w_i, b_i, c_i.

As illustrated in Figure 3.2, the architecture typically leverages a CNN backbone (such as VGG16 or ResNet) to extract rich feature maps, which are then processed by the NetVLAD layer.

The advantages of NetVLAD in visual localization are multifold:

- Robustness: The aggregated descriptors are highly resilient to variations in viewpoint, illumination, and scale. This robustness makes NetVLAD particularly well-suited for place recognition and localization tasks where environmental conditions can vary significantly.

- Efficiency: By producing compact global descriptors, NetVLAD enables fast nearest-neighbor searches within large-scale databases. This efficiency is crucial for real-time applications, such as autonomous navigation and augmented reality.

- End-to-End Training: The differentiable nature of the NetVLAD layer allows the entire network to be fine-tuned on domain-specific data. This optimization improves retrieval performance by directly aligning the learned descriptors with the target task.

In our visual localization pipeline, NetVLAD serves as one of the two primary global feature extractors within a fusion strategy. Its descriptors provide a reliable baseline for capturing the overall scene context, which is essential for both database pair matching and query localization. By combining NetVLAD with complementary techniques, such as AP-GeM and local feature extraction method, we achieve a robust, multi-level representation that enhances both precision and recall in the retrieval process.

Recent studies have demonstrated that NetVLAD achieves state-of-the-art performance on large-scale place recognition benchmarks and has become a cornerstone in many modern visual localization systems [52, 53, 57, 55, 56]. While it excels in handling moderate environmental variations, its performance can be further enhanced when integrated with local feature matching and other complementary descriptors, providing a balanced approach to tackle the inherent challenges of visual localization.

3.1.2 AP-GeM

AP-GeM is a refined variant of the Generalized Mean (GeM) pooling method, developed by Revaud et al. in 2019 [57], and it is specifically optimized for image retrieval tasks. Its key innovation lies in the integration of a loss function that directly optimizes average precision, a metric that aligns closely with the retrieval task, resulting in improved ranking quality of the retrieved images.

GeM pooling exploits the generalized mean to extract one single value from each feature map. In particular, given a CNN output with shape $W \times H \times D$, it processes it as D feature maps $\{x_1, ..., x_D\}$ with dimensions $W \times H$. The pooling results in a D-dimensional vector $f = [f_1, ..., f_D]$, where

$$f_k = \left(\frac{1}{WH} \sum_{x \in x_k} x^{p_k} \right)^{\frac{1}{p_k}}. \tag{3.3}$$

Note that max pooling [58, 59] and average pooling [60] are special cases when $p_k \to \infty$ and $p_k = 1$, respectively. The parameter p_k is learnable, since this pooling is differentiable, allowing for backpropagation.

The introduction of AP-GeM has been influential in the field, setting a new benchmark for learned image representations in retrieval and place recognition. Its development has spurred further research into task-specific pooling techniques and

loss functions, establishing it as a critical component in modern visual localization pipelines.

Historically, while traditional pooling methods (including standard GeM) have been effective for aggregating convolutional features into a single global descriptor, they were not explicitly tailored for ranking performance. AP-GeM addresses this gap by fine-tuning the pooling process with a task-specific objective, thereby producing descriptors that are more discriminative for retrieval scenarios. This task-specific optimization ensures that subtle differences in visual content are better captured, leading to superior performance in challenging conditions.

The main advantages of AP-GeM include:

- Task-Specific Optimization: By directly optimizing average precision, AP-GeM generates descriptors that are finely tuned for ranking tasks, making them particularly effective in distinguishing between similar images.
- Complementarity: When combined with NetVLAD descriptors, AP-GeM contributes complementary information. This fusion leverages the robust global context captured by NetVLAD with the refined, ranking-optimized details provided by AP-GeM, enhancing overall retrieval accuracy.
- Scalability: The global representations produced by AP-GeM are compact and efficient, enabling fast nearest-neighbour searches even when applied to large-scale databases, a critical requirement for real-time visual localization applications.

In our fusion-based approach, AP-GeM works alongside NetVLAD to form a robust global representation. This dual strategy leverages the strengths of both methods: NetVLAD offers a resilient and broadly informative descriptor, while AP-GeM hones in on ranking accuracy and discriminative detail. Together, they enhance both precision and recall in the retrieval process, making the combined system highly effective for challenging visual localization tasks.

3.1.3 SALAD

Sinkhorn Algorithm for Locally Aggregated Descriptors (SALAD) [61] is a novel module built upon the NetVLAD framework, designed to enhance feature assignment and aggregation in visual place recognition. Unlike traditional methods, SALAD integrates a fine-tuned DINOv2 backbone and introduces key modifications to the assignment and aggregation processes.

SALAD utilizes DINOv2, a vision transformer (ViT)-based model, for local feature extraction. Instead of relying on fixed feature extraction methods, a supervised training pipeline is adopted, where only the last layers of DINOv2 are fine-tuned. This approach enhances robustness to appearance variations (e.g.,

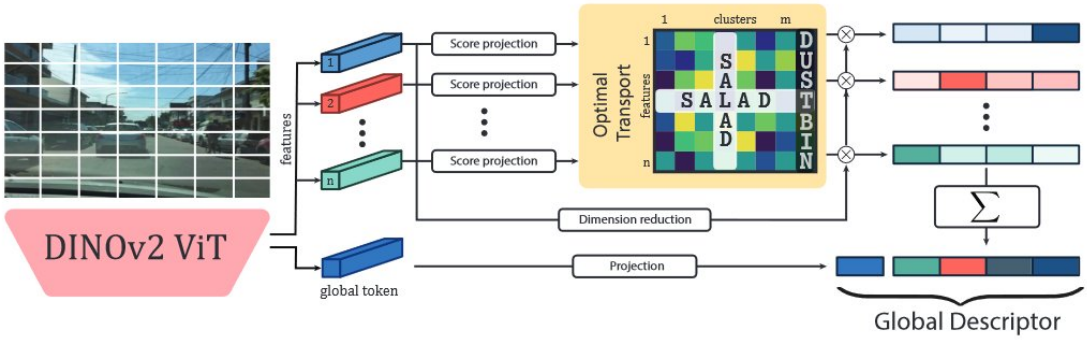


Figure 3.3: First, the DINOv2 backbone extracts local features and a global token from an input image. Then, a small MLP, score projection, computes a score matrix for feature-to-cluster and dustbin relationships. The optimal transport module uses the Sinkhorn algorithm to transform this matrix into an assignment, and subsequently, dimensionality-reduced features are aggregated into the final descriptor based on this assignment and concatenated with the global token. Figure and description from [61]

seasonal and lighting changes) while preserving essential structural information. DINOv2 divides an image $I \in \mathbb{R}^{h \times w \times c}$ into $p \times p \times c$ patches, where $p = 14$. These patches are projected through transformer blocks, yielding tokens $\{t_1, \ldots, t_n, t_{n+1}\}$, with $n = \frac{hw}{p^2}$, where t_{n+1} serves as a global token aggregating class information.

SALAD redefines feature assignment by addressing three key issues in NetVLAD:

1. Instead of initializing cluster centroids using k-means, each row s_i of the assignment score matrix S is learned from scratch using two fully connected layers:
$$s_i = W_{s2}\left(\sigma(W_{s1}t_i + b_{s1})\right) + b_{s2} \tag{3.4}$$
where $W_{s1}, W_{s2}, b_{s1}, b_{s2}$ are the weights and biases, and σ is a non-linear activation function.

2. Features containing negligible information, such as sky regions, are assigned to a 'dustbin' cluster. The score matrix is augmented to $\bar{S} \in \mathbb{R}^{n \times (m+1)}$, incorporating a learnable parameter z for the dustbin assignment:
$$\bar{s}_{i,m+1} = z1_n \tag{3.5}$$
where 1_n is an n-dimensional vector of ones.

3. Unlike NetVLAD's per-row softmax, assignment is formulated as an optimal transport problem, distributing feature mass $\mu = 1_n$ among clusters and the dustbin $\kappa = [1_m^T, n - m]^T$. The Sinkhorn algorithm is applied to normalize row and column distributions in the assignment matrix $\bar{P}$, ensuring:
$$\bar{P}1_{m+1} = \mu, \quad \bar{P}^T 1_n = \kappa \tag{3.6}$$

The dustbin column is removed to obtain the final assignment matrix P.

Once assignment is completed, aggregation is performed using the following modifications:

1. Feature dimensions are reduced from $\mathbb{R}^d$ to $\mathbb{R}^l$ using fully connected layers:

$$f_i = W_{f2}\left(\sigma(W_{f1}t_i + b_{f1})\right) + b_{f2} \tag{3.7}$$

2. Each feature is aggregated into its assigned cluster, without subtracting centroids, as follows:

$$V_{j,k} = \sum_{i=1}^{n} P_{i,k} f_{i,k} \tag{3.8}$$

   where $V \in \mathbb{R}^{m \times l}$ represents the VLAD descriptor.

3. A global descriptor g is extracted from DINOv2's global token:

$$g = W_{g2}\left(\sigma(W_{g1}t_{n+1} + b_{g1})\right) + b_{g2} \tag{3.9}$$

   This vector is concatenated with the flattened V, followed by L2 intra-normalization and global L2 normalization, forming the final descriptor.

3.2 Local Features Extraction and Matching

Local features provide complementary information to global descriptors by capturing fine-grained image details, which are especially useful in distinguishing between visually similar scenes. To achieve this, we decided to use SuperPoint and LightGlue due to their reliability and efficiency. SuperPoint provides robust keypoint detection and description, while LightGlue ensures efficient and accurate feature matching, making them well-suited for our visual localization pipeline.

3.2.1 SuperPoint

SuperPoint [15] is a deep learning-based model designed for keypoint detection and description, offering a powerful alternative to traditional handcrafted methods like SIFT [44], ORB [48], and SURF [62]. By leveraging a self-supervised learning approach, SuperPoint can extract distinctive and repeatable keypoints, making it highly robust to variations in lighting, viewpoint, and occlusions. It has gained widespread use in applications such as visual localization, structure-from-motion, and image retrieval, where reliable feature detection is critical.

Traditional keypoint detection methods rely on handcrafted heuristics, which, while effective, often struggle in complex real-world scenarios. In contrast, SuperPoint utilizes a fully convolutional neural network that learns to detect and

describe keypoints directly from data. This end-to-end design eliminates the need for separate detection and description stages, allowing for a streamlined and efficient feature extraction pipeline.

The architecture of SuperPoint consists of two main components: a keypoint detector and a descriptor generator. The keypoint detection module processes an input image to produce a heatmap that highlights the most salient feature points. Meanwhile, the descriptor module extracts high-dimensional representations for each detected keypoint, ensuring reliable matching across different images. This unified approach significantly improves both speed and accuracy compared to traditional pipelines.

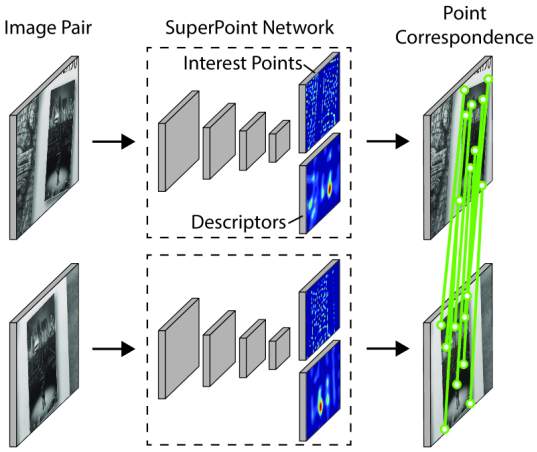


Figure 3.4: SuperPoint is a fully-convolutional neural network that computes SIFT-like 2D interest point locations and descriptors in a single forward pass. Figure from [15]

A major innovation of SuperPoint lies in its self-supervised training strategy, which enables it to learn without manually labeled keypoints. Initially, the model is trained using synthetic data, where keypoints are generated by applying simple homographic transformations. This provides an initial learning signal for detecting meaningful feature points. Following this pretraining phase, the model is refined on real-world data using a bootstrapping approach, where it leverages its own predictions to improve accuracy iteratively. This adaptive learning process allows SuperPoint to generalize well to diverse environments without requiring extensive human annotation.

One of SuperPoint's biggest strengths is its robustness to challenging conditions such as changes in illumination, perspective distortion, and occlusions. Unlike traditional handcrafted methods, which may fail in such scenarios, SuperPoint's learned features allow it to maintain strong performance. Its fully convolutional architecture also enables real-time processing, making it suitable for applications like robotic perception, augmented reality, and autonomous navigation. By combining

detection and description within a single deep learning framework, it offers an efficient and scalable solution for various vision tasks.

Despite its advantages, SuperPoint is not without limitations. Its computational cost, while lower than some deep learning alternatives, is still higher than traditional methods such as FAST [63] or ORB [48]. Additionally, while it generalizes well to many conditions, extreme distortions or highly dynamic environments can still pose challenges.

3.2.2 LightGlue

LightGlue [64] is a deep learning-based feature matching framework designed to provide a balance between efficiency and accuracy, addressing the limitations of traditional feature matching approaches [65, 66]. As an evolution of transformer-based architectures [14, 67, 68], LightGlue introduces an adaptive matching strategy that dynamically refines correspondences based on the complexity of the scene. This makes it particularly well-suited for real-time applications in robotics, augmented reality, and large-scale structure-from-motion pipelines.

Traditional feature matching methods, such as brute-force nearest neighbor search or handcrafted algorithms like FLANN [65], often suffer from inefficiencies when dealing with large feature sets. These approaches apply a fixed matching strategy regardless of image complexity, leading to unnecessary computations in simple cases or insufficient accuracy in more challenging scenarios. LightGlue overcomes these limitations by introducing an adaptive approach that adjusts the number of iterations and computational resources based on scene difficulty.

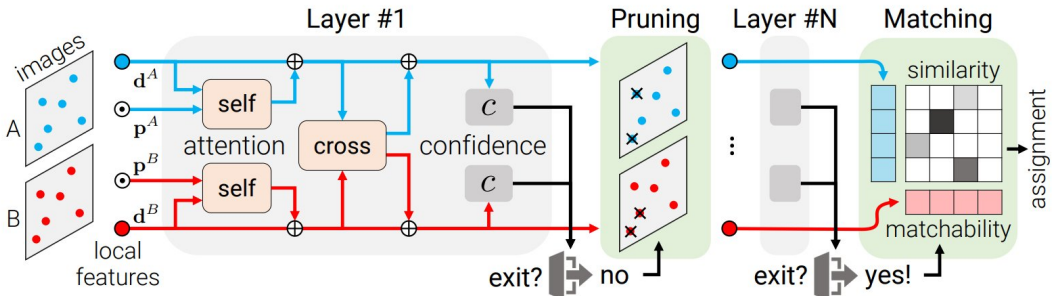


Figure 3.5: Given a pair of input local features (d, p), each layer augments the visual descriptors (•, •) with context based on self- and cross-attention units with positional encoding ⊙. A confidence classifier c helps decide whether to stop the inference. If few points are confident, the inference proceeds to the next layer but we prune points that are confidently unmatchable. Once a confident state if reached, LightGlue predicts an assignment between points based on their pariwise similarity and unary matchability. Image and description from [64]

The core of LightGlue's architecture is built upon a transformer-based attention mechanism that enables context-aware feature matching. Unlike traditional models that treat all keypoints equally, LightGlue prioritizes high-confidence matches early in the process, dynamically refining lower-confidence matches as needed. This hierarchical refinement allows it to efficiently process simple scenes while dedicating more resources to challenging ones, ensuring an optimal trade-off between speed and accuracy.

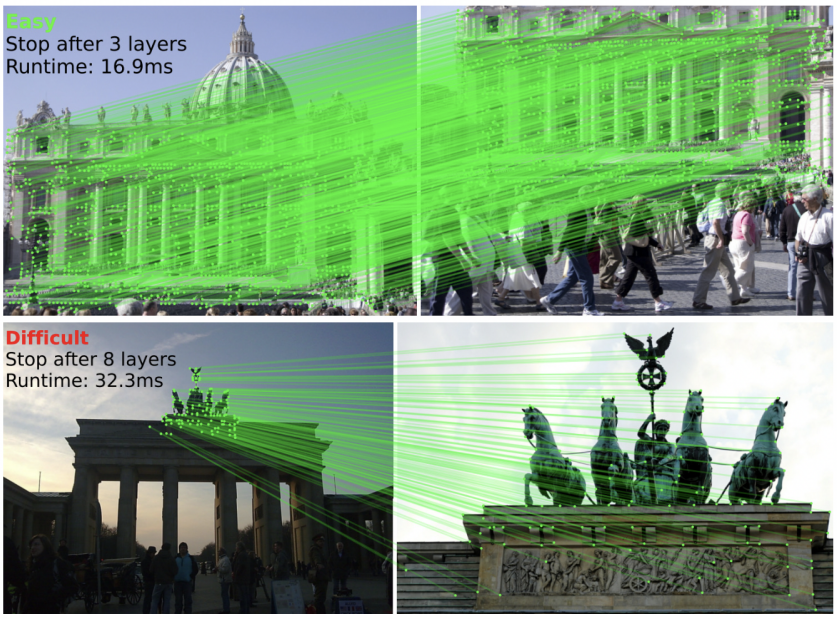


Figure 3.6: LigthGlue matching an easy image pairs (top) and a difficult ones (bottom). Image from [64]

One of the key innovations of LightGlue is its ability to leverage a coarse-to-fine matching strategy. Initially, it establishes rough correspondences between keypoints, rapidly filtering out obvious mismatches. Once a preliminary set of correspondences is established, the model refines the matches using self- and cross-attention mechanisms, progressively improving alignment accuracy. This stepwise refinement makes LightGlue more robust to occlusions, viewpoint changes, and varying image conditions compared to conventional methods.

In addition to its efficiency, LightGlue is designed with flexibility in mind. It can integrate seamlessly with different feature extractors, such as SuperPoint or SIFT, adapting its performance to the available keypoint descriptors. This modular design ensures that it can be deployed in a wide range of applications without requiring a complete overhaul of existing vision pipelines. Furthermore, its lightweight nature allows it to run efficiently on edge devices, making it ideal for mobile and embedded

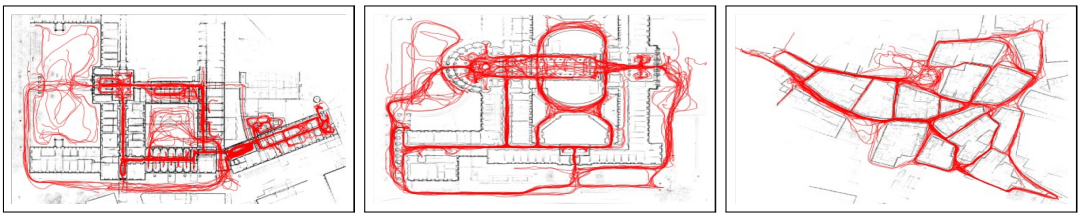
systems where computational resources are limited.

Despite its advantages, LightGlue does have some limitations. While it significantly improves efficiency compared to traditional deep learning-based feature matchers, it still relies on transformer-based computations, which can be more resource-intensive than purely handcrafted approaches. Additionally, its performance is highly dependent on the quality of the initial keypoint detections, meaning that suboptimal feature extraction can impact the final matching results.

3.3 Datasets

To evaluate the performance of our retrieval-based localization methods, we conducted experiments using two established datasets: Lamar and VBR. These datasets provide realistic and diverse visual data, making them suitable for benchmarking retrieval and localization approaches.

3.3.1 LaMAR

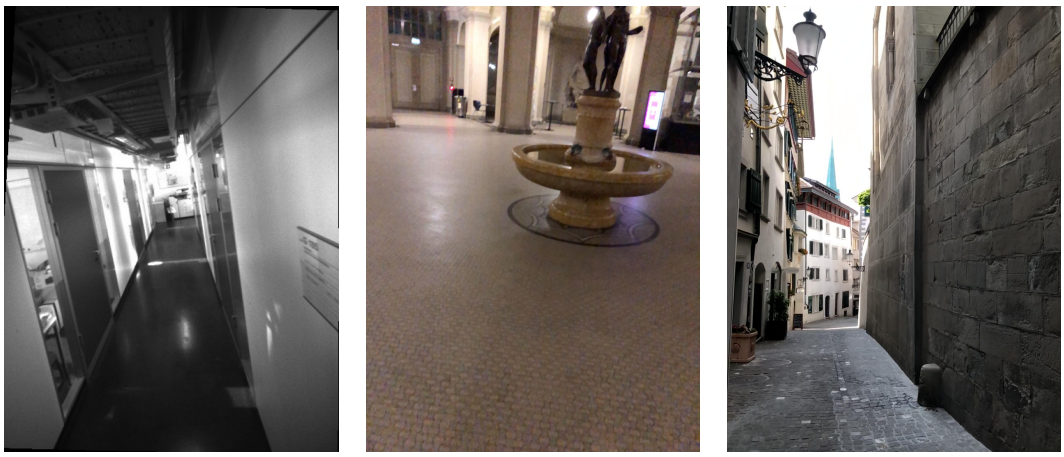


(a) CAB **(b)** HGE **(c)** LIN

Figure 3.7: Summary of LaMAR sequences. Images from [45].

The LaMAR Dataset [45] is a large-scale dataset created to support research in visual localization and mapping, with a focus on urban and suburban environments and augmented reality applications. The initial release of LaMAR comprises three expansive locations that serve as representative scenarios for AR use cases. Specifically, HGE covers approximately 18,000 m^2 of a historical university building's ground floor, featuring multiple large halls and expansive esplanades; CAB spans around 12,000 m^2 and consists of a multi-floor office building with a variety of small and large offices, a kitchen, storage rooms, and two courtyards; and LIN encompasses about 15,000 m^2, capturing several blocks of an old town with shops, restaurants, and narrow passages. Both HGE and CAB include extensive indoor and outdoor sections with numerous symmetric structures, and each location has undergone structural changes over the span of a year (e.g., the front of HGE transforming into a construction site or rearrangements of indoor furniture).

Data for the LaMAR Dataset was collected in a crowd-sourced fashion using consumer-grade devices, primarily Microsoft HoloLens 2 and Apple iPad Pro, equipped with custom raw sensor recording applications. Ten participants were instructed to freely explore the designated areas, resulting in diverse camera heights, motion patterns, and capture conditions. Recordings were performed both during the day and at night over the course of up to one year, with each location being covered by more than 100 sessions of roughly 5 minutes each. In addition, each location was captured two to three times by professional mapping platforms (such as the NavVis M6 trolley or VLX backpack), which generate textured dense 3D models from laser scanner data and panoramic imagery. To adhere to privacy regulations, the data processing pipeline automatically anonymizes all visible faces and license plates.



(a) CAB **(b)** HGE **(c)** LIN

Figure 3.8: Three example query images from the Lamar dataset, respectively they are from CAB, HGE and LIN.

The dataset consists of high-quality ground-level images acquired under a wide range of conditions, including variations in weather, lighting, and seasons. This diversity offers a challenging benchmark for evaluating algorithms in scenarios characterized by significant viewpoint differences and cross-domain variations. Each image is enriched with precise metadata, such as GPS coordinates, camera poses, and ground truth information, which is derived via an automated pipeline that registers the AR sequences to a dense 3D reference model constructed from LiDAR scans. This pipeline involves pairwise registration, global alignment, and pose graph optimization, yielding accurate and globally consistent absolute poses in a common reference frame. Temporal variations, achieved by repeated visits over different periods, further enable the study of long-term environmental changes.

dataset	out/indoor	changes	scale	density	camera motion	imaging devices	additional sensors	ground truth	accuracy
Aachen [67,66]	✓ ✗		★★½	★★☆	still images	DSLR	✗	SfM	>dm
Phototourism [34]	✓ ✗		★☆☆	★★★	still images	DSLR, phone	✗	SfM	∼m
San Francisco [14]	✓ ✗		★★★	★☆☆	still images	DSLR, phone	GNSS	SfM+GNSS	∼m
Cambridge [37]	✓ ✗		★☆☆	★★☆	handheld	mobile	✗	SfM	>dm
7Scenes [73]	✗ ✓	✗	★☆☆	★★★	handheld	mobile	depth	RGB-D	∼cm
RIO10 [84]	✗ ✓		★☆☆	★★★	handheld	Tango tablet	depth	VIO	>dm
InLoc [77]	✗ ✓		★★☆	★☆☆	still images	panoramas, phone	lidar	manual+lidar	>dm
Baidu mall [76]	✗ ✓		★★☆	★★☆	still images	DSLR, phone	lidar	manual+lidar	∼dm
Naver Labs [40]	✗ ✓		★★☆	★★☆	robot-mounted	fisheye, phone	lidar	lidar+SfM	∼dm
NCLT [12]	✓ ✓		★★☆	★★☆	robot-mounted	wide-angle	lidar, IMU, GNSS	lidar+VIO	∼dm
ADVIO [57]	✓ ✓		★★☆	★☆☆	handheld	phone, Tango	IMU, depth, GNSS	manual+VIO	∼m
ETH3D [71]	✓ ✓	✗	★☆☆	★★☆	handheld	DSLR, wide-angle	lidar	manual+lidar	∼mm
LaMAR (ours)	✓ ✓		★★½ 3 locations 45'000 m^2	★★★ 100 hours 40 km	handheld head-mounted	phone, headset backpack, trolley	lidar, IMU, depth, infrared	lidar+SfM+VIO automated	∼cm

Figure 3.9: Characteristic of the Lamar dataset compared to other famous datasets. Figure from [45].

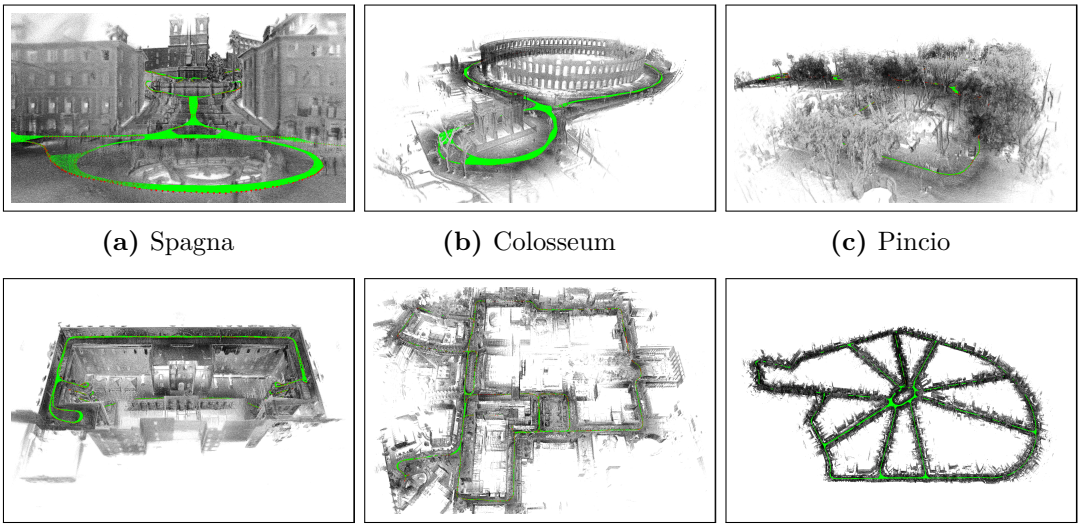
In addition to visual data, the LaMAR Dataset includes multimodal sensor information, such as depth maps, LiDAR scans, and IMU readings, making it highly versatile for applications like SLAM, visual odometry, and real-time localization. However, for the purposes of our research, due to the unavailability of ground truth for the official test set, the validation set was repurposed as the testing set. Moreover, given that the focus of this research is the role of image retrieval, only smartphone-acquired and hololens-acquired validation queries were used, and no additional sensor modalities (e.g., IMU, Wi-Fi, Bluetooth signals, depth, or infrared) were incorporated.

This decision does not affect the integrity of our experiments since our study is comparative rather than absolute. Our primary objective is to compare different retrieval methods, modalities, and configurations rather than establish an absolute performance measure. By using the validation set as the test set, we maintain a consistent and reliable evaluation framework.

3.3.2 VBR: Vision Benchmark in Rome

The Visual Benchmark in Rome (VBR) [46] dataset was introduced to evaluate visual recognition and localization models in complex urban environments, with a particular emphasis on Rome's rich historical and architectural heritage. VBR is a large-scale dataset offering extensive coverage of the city, capturing its iconic landmarks and diverse urban fabrics under a variety of conditions.

A notable strength of VBR is its sophisticated data acquisition system. The dataset was collected using a multi-modal sensor suite comprising two global shutter RGB cameras arranged in a wide stereo configuration, a high-resolution 3D



(a) Spagna **(b)** Colosseum **(c)** Pincio

(d) DIAG **(e)** Campus **(f)** Ciampino

Figure 3.10: Summary of VBR sequences. Images from[46].

LiDAR, an RTK-GPS system, and an Inertial Measurement Unit (IMU). Rigorous calibration and synchronization of these sensors ensure that the resulting ground truth, expressed in the precise LiDAR reference frame, achieves accuracies on the order of a few centimeters, even over extensive trajectories.

The dataset is structured to rigorously challenge visual localization algorithms. The reference database contains high-quality images captured under controlled conditions, providing a stable basis for training and benchmarking. In contrast, the query images are recorded from varied viewpoints and at different times, featuring substantial variations in lighting, weather, and crowd density. This deliberate dual mode acquisition, controlled references versus unconstrained queries, forces models to contend with real-world cross-domain discrepancies and ensures a comprehensive evaluation of robustness and accuracy.

The VBR dataset is originally split into a train set and a test set. However, as with Lamar, the test set does not contain ground truth positions, preventing us from computing localization accuracy. To work around this, we carefully selected 750 images for each scene from the train set to construct our test set.

To ensure meaningful evaluation, we sampled images from specific trajectory segments where the camera operator passed through multiple times. This allowed us to create a test set where nearby and visually similar images remain in the training set, ensuring a realistic retrieval scenario. The remaining images from the train set were used to construct the global map. The specific timestamps of the selected test images are listed in Table 3.1.

However, even after our careful selection process, this approach introduced

several issues that affect the validity of our experiments on this dataset. These limitations and their impact on our results will be discussed in detail in Chapter 4.6.

Scene	**Timestamp of test images**
Campus	1273791462170 - 1324036965410
Ciampino	3333899690710 - 3361796843540
	4320655133829 - 4394998052910
Colosseo	235383958310 - 2442227787890
Diag	2824193513930 - 3062070457450
Pincio	284343555140 - 707687060340
Spagna	213141675270 - 460611290830

Table 3.1: Timestamp of test images for VBR

Chapter 4

Experiments

In this chapter, we present a comprehensive evaluation of our experiments on the visual localization pipeline. Our experimental framework is designed to isolate the impact of the image retrieval phase, during query time, on pose estimation performance by keeping all other components of the pipeline constant. Specifically, we use the same structure-from-motion map, and local feature extraction and matching procedures across all experiments. This controlled setup ensures that any differences in localization accuracy are solely due to variations in the image retrieval strategy.

We begin by defining the evaluation metrics, Recall@[1, 0.1] and Recall@[5, 1], which capture the system's ability to correctly localize query images under both stringent and relaxed error thresholds.

The experiments in this chapter are organized into three main parts:

1. **Experiment 1: Evaluating the Role of Angular Differences Between Retrieved Images.** In this experiment, we investigate whether increasing the angular diversity among the images retrieved during query time improves the final pose estimation. We compare a baseline retrieval method (selecting the first n images from a spatially filtered list) against an approach that maximizes angular spread among the retrieved images.

2. **Experiment 2: Model Comparison (NetVLAD & AP-GeM vs. SALAD).** Here, we compare the performance of two different global feature extractors, one fusion-based (NetVLAD combined with AP-GeM) and the other SALAD-based. For each query image, we generate a ranked list of the top 40 database images and evaluate three retrieval strategies: a baseline using the top 10 images, a method using all 40 images, and a third strategy that leverages local features for additional refinement. An upper bound for every scene is also established.

3. **Experiment 3: Incorporating Local Features to Improve Retrieval.** In the final experiment, we propose an advanced retrieval method that incorporates local feature information to further refine the selection of images. For each query, we first generate a ranked list of 40 images based on global feature similarities. We then cluster these images using the DBSCAN algorithm, tuning the parameters based on the known positions of the database images, and select the best cluster based on the maximum ratio of matching local features. From this cluster, we retrieve at most 10 images for pose estimation.

The following sections detail the methodology and results for each experiment.

4.1 Metrics for Evaluation

In this work, the primary metric for evaluating visual localization performance is Recall, measured under two sets of error thresholds: Recall@[1, 0.1] and Recall@[5, 1]. These metrics assess the system's ability to correctly localize query images by considering both angular and positional accuracy.

Definition of Metrics

- Recall@[1, 0.1]: A query is deemed correctly localized if the pose estimation result exhibits an orientation error of no more than 1 degree and a positional error of no more than 0.1 meters. This metric reflects the system's performance under stringent accuracy requirements.

- Recall@[5, 1]: A query is considered successfully localized if the pose estimation result has an orientation error of no more than 5 degrees and a positional error of no more than 1 meter. This more lenient metric captures the system's performance when higher error tolerances are acceptable.

These two metrics together provide a comprehensive view of the visual localization system's performance under both strict and relaxed conditions.

The recall metrics Recall@[1, 0.1] and Recall@[5, 1] have been adopted from the Lamar paper.

Visual localization systems must achieve high accuracy in both orientation and position to be effective in real-world applications. The chosen thresholds are designed to reflect critical operational requirements: The Recall@[1, 0.1] metric assesses the system under conditions demanding very high precision, which is vital for applications where even minor deviations can have significant consequences. The Recall@[5, 1] metric evaluates the system under more relaxed conditions, which can be useful in scenarios where a broader tolerance for error is acceptable.

4.2 Fixed Pipeline Components for Isolating Retrieval Impact

To isolate and better understand the influence of the image retrieval process during query time on pose estimation performance, we ensured that all other components of the visual localization pipeline were held constant across our experiments. By doing so, any observed differences in results can be directly attributed to variations in the retrieval step during query time. This controlled experimental setup includes the following:

- **Local Feature Extraction:** The parameters for local feature extraction are identical for both database and query images:
  - Method: SuperPoint
  - Maximum keypoints:2048
  - NMS radius: 3
  - Grayscale: True
  - Maximum resize: 1024
- **Global Feature Extraction:** For global feature extraction, the same settings were applied to both database and query images:
  - Methods: SALAD, NetVLAD, and AP-GeM
  - Resize maximum: 640
  - Additional filters:
    - Filter frustum: True
    - Filter pose: True
    - Number of pairs filter: 250
- **Structure-from-Motion Map:** For every scene, an initial SfM map was built retrieving 10 images from the database per image. As described in Section 3.2, SuperPoint was used to extract local features and LightGlue to match them.
- **Local Feature Matching (Map-Map):** For matching features between images in the map, preprocessing is performed using:
  - Grayscale: True
  - Resize maximum: 1024

By maintaining these fixed parameters across all experiments, we ensure that any variations in localization performance are solely due to the changes in the image retrieval strategy during query time.

4.3 Experiment 1: Evaluating the Role of Angular Differences Between Retrieved Images

In this experiment, we aim to gain insights into how the diversity of viewpoints in the image retrieval phase affects the final camera pose estimation. In particular, we investigate whether a wider spread in the angular differences between the retrieved images, those that are subsequently used to estimate the query's pose, can lead to improved localization accuracy.

The experiment is designed to compare two retrieval strategies: one based on SALAD and another using Netvlad + APGem global feature extractors. For each query image, of each scene, we first generate a ranked list of the top 40 database images by computing global feature similarities. From this list, we apply a spatial filter to retain only those images that lie within a specified range from the query. This filtering ensures that the images considered are likely to be relevant to the query.

As described in section 4.2, all other components of the pipeline remained constant.

Two sets of experiments are then conducted:

1. **Baseline Retrieval:** From the filtered list, for each query, we take the first n images (with n varying from 3 to 10) as the set of images used for pose estimation.

2. **Angular Diversity Maximization:** We further refine the selection by choosing n images that not only satisfy the spatial filter but also maximize the angular spread. Specifically, we first select the top image in terms of global feature similarity. Then, from the remaining images in the filtered set, we iteratively choose the one that maximizes angular diversity relative to the images already selected. This process is repeated until n images have been chosen. This approach is intended to capture a more diverse set of viewpoints, while still ensuring relevance to the query.

   Formally, let Q be the query image, $S = \{I_1, I_2, ..., I_m\}$ be the set of candidate images filtered by spatial constraints, $f(I)$ be the global feature descriptor of an image I, $\theta(I, J)$ be the angular difference between the viewpoints of images I and J, n be the number of images to select and T be the final selected subset of n images.

   The first image I_1 is chosen based on the highest global feature similarity with the query Q:

$$I_1 = \arg\max_{I \in S} \text{sim}(f(I), f(Q))$$

where $\text{sim}(\cdot, \cdot)$ is a similarity function, such as cosine similarity.

For each subsequent selection I_k $(k = 2, ..., n)$, we choose the image from the remaining set S_k that maximizes the minimum angular difference from the already selected images:

$$I_k = \arg\max_{I \in S_k} \min_{J \in T} \theta(I, J)$$

where $S_k = S \setminus T$ is the remaining candidate set after previous selections and T is the set of already selected images $\{I_1, ..., I_{k-1}\}$.

So, the final formula representation is the following:

$$T = \{I_1\} \cup \left\{ \arg\max_{I \in S_k} \min_{J \in T} \theta(I, J) \quad \forall k = 2, ..., n \right\}$$

In both cases, filtering images to lie within a certain range from the query is a critical step. This constraint guarantees that when we maximize the angular spread, we are likely still selecting images that contain the same scene as the query, thereby avoiding the inclusion of outliers that might degrade pose estimation accuracy.

By comparing the pose estimation results obtained with the baseline retrieval against those achieved with the angular diversity maximization strategy, and doing so for varying numbers of retrieved images, we can assess the impact of viewpoint diversity on the overall performance of the localization pipeline.

Figures 4.1 and 4.2 display the experimental results on the Lamar dataset, illustrating the performance differences between the various retrieval strategies. In these graphs, the R@(1, 0.1) metric is plotted against the number of retrieved images, comparing both fusion-based and SALAD-based global feature extractors, as well as the baseline retrieval method versus the angular spread selection approach. These visual comparisons make it easy to appreciate how performance varies with the number of images and to discern the differences between the methods.

Preliminary observations indicate that the SALAD-based global feature extractor consistently outperforms the fusion-based extractor on the Lamar dataset under the tested conditions. In particular, for phone queries for CAB and HGE scenes, the angular spread method generally provides higher R@(1, 0.1) scores compared to the baseline approach. In contrast, for the phone queries for LIN scene, while SALAD again outperforms the fusion-based extractor, the performance differences between the baseline and angular spread methods are minimal. Instead, for the hololens queries no difference is appreciable apart from few points. These trends are further explored in detail in Chapter 5.

For more detailed results for this experiment, including additional results for the R@(5, 1) metric, refer to Tables A.1, A.2 and A.3 in Appendix A.

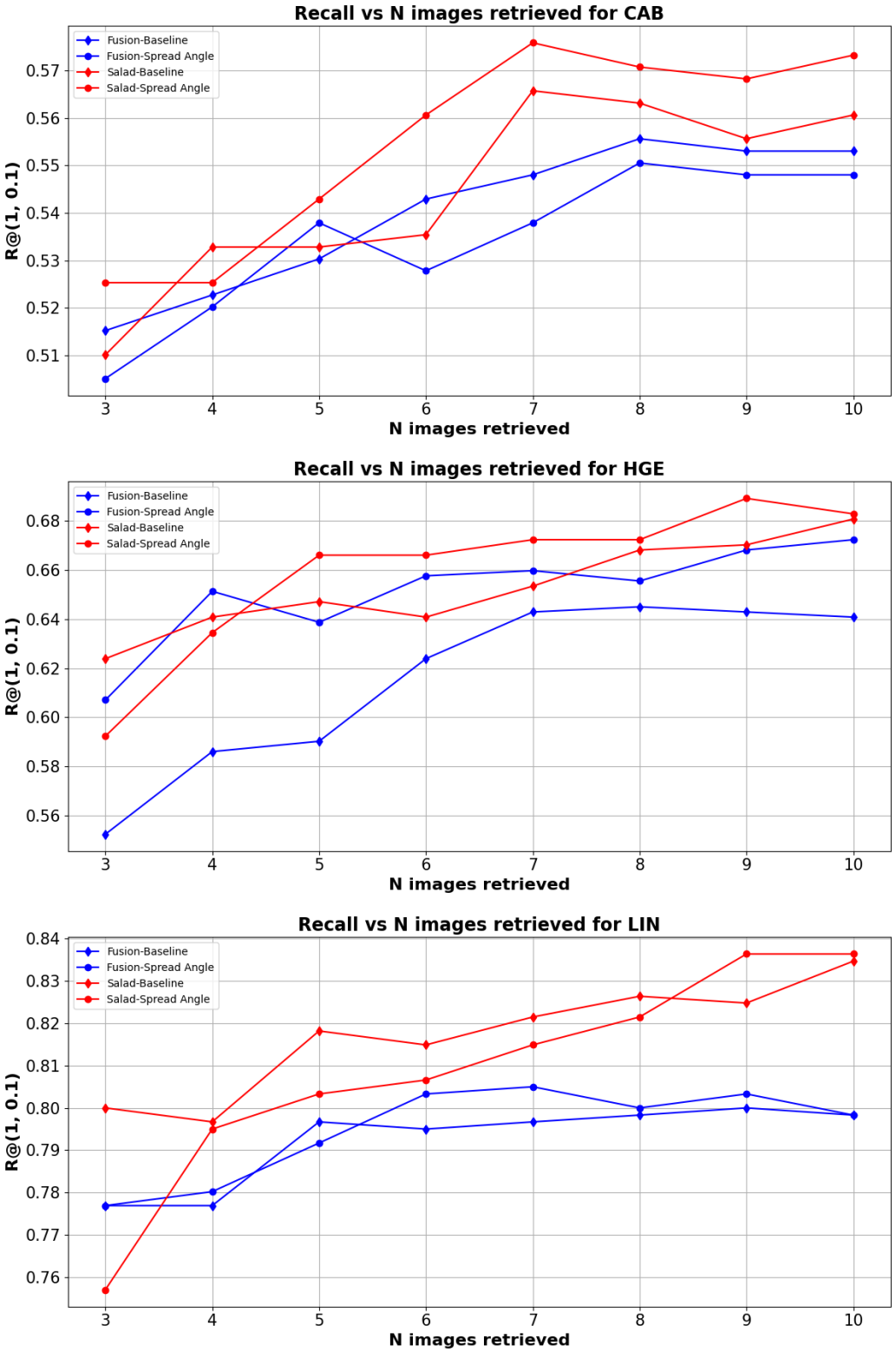


Figure 4.1: Impact of retrieval strategy on pose estimation Performance for the Lamar Dataset for phone queries. Each panel corresponds to a different scene (CAB, HGE, and LIN) where the x-axis represents the number of retrieved images and the y-axis indicates the R@(1, 0.1) metric. Four curves are shown for each scene: Fusion-based and SALAD-based retrieval using the baseline strategy, and Fusion-based and SALAD-based retrieval with angular spread maximization.

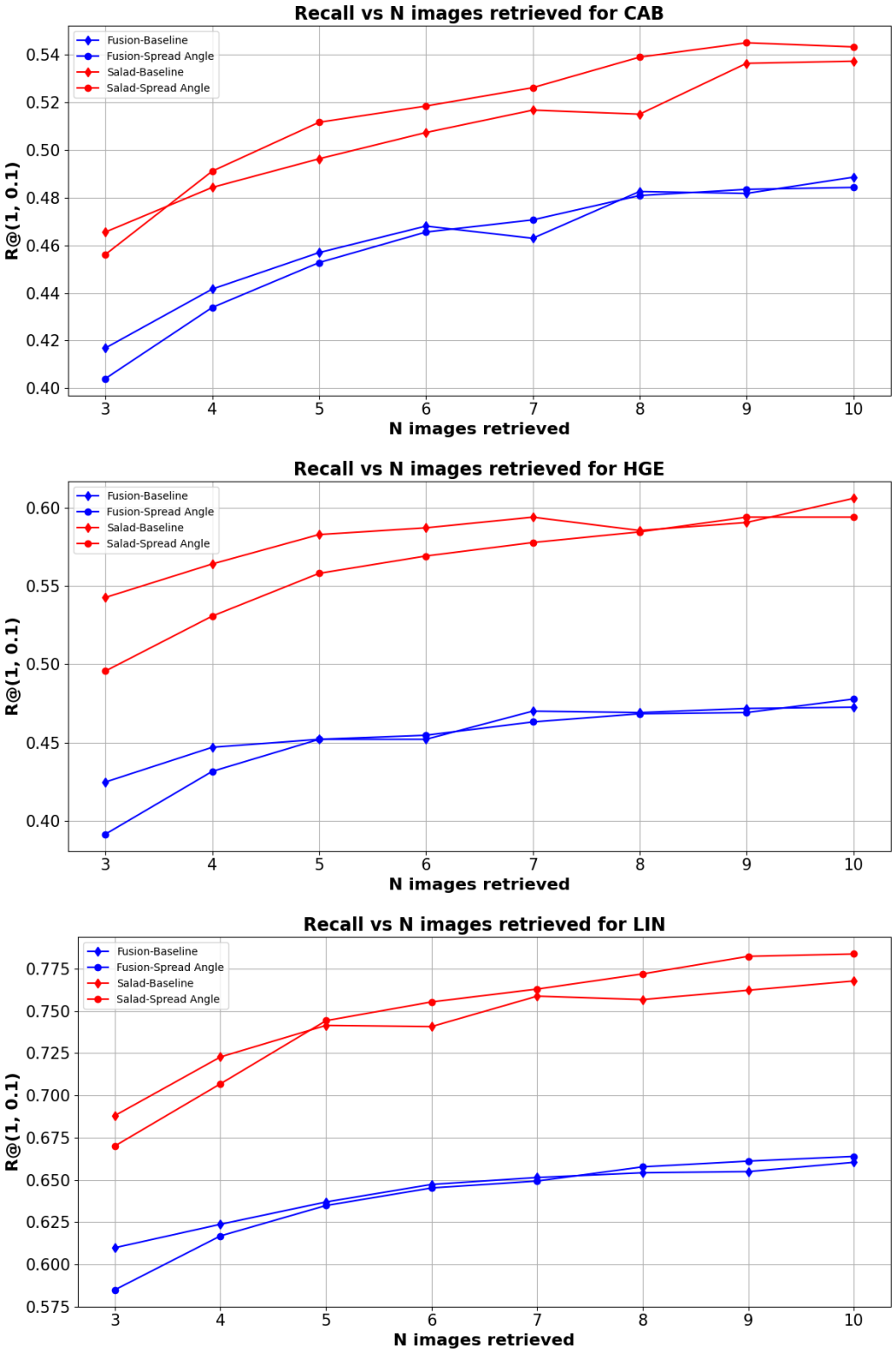


Figure 4.2: Impact of retrieval strategy on pose estimation Performance for the Lamar Dataset for hololens queries. Each panel corresponds to a different scene (CAB, HGE, and LIN) where the x-axis represents the number of retrieved images and the y-axis indicates the R@(1, 0.1) metric. Four curves are shown for each scene: Fusion-based and SALAD-based retrieval using the baseline strategy, and Fusion-based and SALAD-based retrieval with angular spread maximization.

4.4 Experiment 2: Model Comparison (NetVLAD & AP-GeM vs. SALAD)

In this experiment, we compare the performance of two different global feature extractors: a fusion-based approach (combining NetVLAD and AP-GeM) versus the SALAD-based extractor. Our goal is to understand how these methods perform on the dataset when applied during the image retrieval phase.

For each query image, for each scene, we first generate a ranked list of the top 40 database images based on global feature similarity. From this ranked list, a baseline result is computed by selecting only the first 10 images for both the fusion-based and SALAD-based extractors. Next, we evaluate the performance when all 40 images are considered, in order to assess how the retrieval performance varies with an increased number of images.

Finally, we establish an upper bound for each scene by increasing the number of images retrieved during map creation to 50. To create these upperbound, during query time, the retrieved images are further filtered by distance, using the ground truth query position, and by imposing a local feature matching criterion (requiring at least 5% of the features to match with the query image).

As in the previous experiment, all other components of the pipeline remained constant. This controlled setup allows us to directly evaluate the impact of the global extractor on image retrieval estimation accuracy.

Dataset	Method	Fusion		SALAD	
Phone		R@(1, 0.1)	R@(5, 1.0)	R@(1, 0.1)	R@(5, 1.0)
CAB	10 Images Retrieved	0.422	0.533	0.439	0.540
CAB	40 Images Retrieved	0.465	0.568	0.523	0.636
HGE	10 Images Retrieved	0.548	0.813	0.664	0.918
HGE	40 Images Retrieved	0.647	0.884	0.681	0.907
LIN	10 Images Retrieved	0.767	0.891	0.8	0.935
LIN	40 Images Retrieved	0.820	0.940	0.833	0.970

Table 4.1: Results for experiment 2 on phone query images, model comparison. The table presents the performance of the two image retrieval methods (baseline, all-40) for both the fusion-based (NetVLAD & AP-GeM) and SALAD-based global extractors. Metrics R@(1,0.1) and R@(5,1.0) are reported, with the best results for each configuration highlighted in red.

The results for this experiment are presented in Tables 4.1 and 4.2. In these tables, we report the R@(1,0.1) and R@(5,1.0) metrics for both the fusion-based and SALAD-based extractors when 10 images and 40 images are retrieved. As

Dataset Hololens	Method	Fusion		SALAD	
		R@(1, 0.1)	R@(5, 1.0)	R@(1, 0.1)	R@(5, 1.0)
CAB	10 Images Retrieved	0.445	0.640	0.508	0.738
CAB	40 Images Retrieved	0.496	0.701	0.558	0.785
HGE	10 Images Retrieved	0.422	0.6	0.539	0.710
HGE	40 Images Retrieved	0.462	0.640	0.577	0.739
LIN	10 Images Retrieved	0.622	0.729	0.753	0.891
LIN	40 Images Retrieved	0.671	0.777	0.793	0.909

Table 4.2: Results for experiment 2 on hololens query images, model comparison. The table presents the performance of the two image retrieval methods (baseline, all-40) for both the fusion-based (NetVLAD & AP-GeM) and SALAD-based global extractors. Metrics R@(1,0.1) and R@(5,1.0) are reported, with the best results for each configuration highlighted in red.

shown, for every scene and across both query types (phone and hololens), the SALAD-based global feature extractor consistently outperforms the fusion-based extractor for both R@(1,0.1) and R@(5,1.0) metrics.

Table 4.3 shows upperbound for the three scenes, for both query and hololens queries, calculated by constructing the map with 50 retrieved images and further filtering by distance using the query ground truth pose and local feature matching ratio.

Dataset	Phone		Hololens	
	R@(1, 0.1)	R@(5, 1.0)	R@(1, 0.1)	R@(5, 1.0)
CAB	0.680	0.823	0.662	0.9
HGE	0.745	0.946	0.707	0.860
LIN	0.881	0.975	0.846	0.943

Table 4.3: Upper bound results for the CAB, HGE, and LIN scenes. The table reports the computed upper bounds for the R@(1,0.1) and R@(5,1.0) metrics, obtained by constructing the map with 50 retrieved images and further filtering by distance (using the query ground truth pose) and local feature matching ratio.

4.5 Experiment 3: Incorporating Local Features to Improve Retrieval

In this experiment, we propose an advanced method to refine the image retrieval process by incorporating local feature information during query time. Our objective is to retrieve at most 10 images per query, highlighting the additional information that local features can contribute to the selection of the best images for pose estimation.

For each query image, a ranked list of the top 40 database images is generated by computing global feature similarities, as in our earlier experiments. To leverage the additional information provided by local features, we then cluster these 40 images using the DBSCAN algorithm. In this clustering process, we utilize the known positions of the database images and carefully tune the DBSCAN parameters, specifically *eps* and the minimum number of samples, to obtain meaningful clusters. After clustering, we select the best cluster based on the maximum ratio of matching local features between the query image and the images in each cluster. From this optimal cluster, we then select at most 10 images for subsequent pose estimation. This approach ensures that the final set of images is not only similar to the query in terms of global features but also exhibits strong local feature correspondences, which can provide additional spatial context and improve the accuracy of the pose estimation.

Formally, for a given query image Q, we first retrieve a ranked set of the top 40 images from the database D based on global feature similarity:

$$S_{40} = \{I_1, I_2, ..., I_{40}\}, \quad \text{where} \quad I_i = \arg\max_{I \in D} \text{sim}(f_g(I), f_g(Q))$$

where $f_g(I)$ represents the global feature descriptor of image I, $\text{sim}(\cdot, \cdot)$ denotes the similarity function (e.g., cosine similarity), S_{40} is the set of top 40 retrieved images.

To refine the retrieved set, we apply DBSCAN clustering to the top 40 images using their known positions. Let each image I_i have a spatial position $p(I_i)$, then we define the clustering process as:

$$\mathcal{C} = \text{DBSCAN}(\{p(I_i)\}_{i=1}^{40}, \epsilon, N_{\min})$$

where: $\mathcal{C} = \{C_1, C_2, ..., C_m\}$ represents the set of clusters, ϵ is the neighborhood radius parameter, $N_{\min}$ is the minimum number of samples required to form a cluster.

From the formed clusters, we select at most 10 images for the final retrieval set S_{10}:

$$S_{10} = \bigcup_{C_k \in \mathcal{C}} \{I_k\}, \quad |S_{10}| \leq 10$$

where the selection strategy prioritizes images from the most relevant clusters.

As in the previous experiments, all other components of the pipeline remained constant.

Dataset	Method	Fusion		Salad	
Phone		R@(1, 0.1)	R@(5, 1.0)	R@(1, 0.1)	R@(5, 1.0)
CAB	10 Images Retrieved	0.422	0.533	0.439	0.540
CAB	40 Images Retrieved	0.465	0.568	0.523	0.636
CAB	CL+LF 10 Images	0.459	0.553	0.469	0.563
HGE	10 Images Retrieved	0.548	0.813	0.664	0.918
HGE	40 Images Retrieved	0.647	0.884	0.681	0.907
HGE	CL+LF 10 Images	0.615	0.846	0.640	0.909
LIN	10 Images Retrieved	0.767	0.891	0.8	0.935
LIN	40 Images Retrieved	0.820	0.940	0.833	0.970
LIN	CL+LF 10 Images	0.760	0.877	0.796	0.902

Table 4.4: Results for experiment 3 on phone query images. The table presents the performance of the three image retrieval methods (baseline, all-40, and local feature-based selection) for both the fusion-based (NetVLAD & AP-GeM) and SALAD-based global extractors. Metrics R@(1,0.1) and R@(5,1.0) are reported.

By comparing the pose estimation results achieved with this local feature-enhanced retrieval method against those from the previous methods, we can assess the impact of incorporating local feature information into the retrieval phase.

An objective evaluation of the results 4.4 and 4.5 shows that our proposed cluster + local features method consistently outperforms the baseline approach that relies on the first 10 retrieved images, even when using an equal or fewer number of images. Anyhow, our method does not surpass the performance of the 40-images retrieved strategy but in several cases it comes close, demonstrating competitive results, particularly in certain scenes and with specific global extractors.

Additionally, the SALAD-based global extractor continue to outperform the fusion-based extractor also in the cluster + local features experiment.

Dataset Hololens	Method	Fusion		Salad	
		R@(1, 0.1)	R@(5, 1.0)	R@(1, 0.1)	R@(5, 1.0)
CAB	10 Images Retrieved	0.445	0.640	0.508	0.738
CAB	40 Images Retrieved	0.496	0.701	0.558	0.785
CAB	CL+LF 10 Images	0.473	0.662	0.522	0.759
HGE	10 Images Retrieved	0.422	0.6	0.539	0.710
HGE	40 Images Retrieved	0.462	0.640	0.577	0.739
HGE	CL+LF 10 Images	0.453	0.633	0.558	0.733
LIN	10 Images Retrieved	0.622	0.729	0.753	0.891
LIN	40 Images Retrieved	0.671	0.777	0.793	0.909
LIN	CL+LF 10 Images	0.653	0.770	0.758	0.892

Table 4.5: Results for experiment 3 on hololens query images. The table presents the performance of the three image retrieval methods (baseline, all-40, and local feature-based selection) for both the fusion-based (NetVLAD & AP-GeM) and SALAD-based global extractors. Metrics R@(1,0.1) and R@(5,1.0) are reported.

4.6 Experiments on VBR

In this chapter, we present a brief overview of the results obtained from three experiments conducted on the Vision Benchmark in Rome (VBR) dataset. However, it is important to note that these results should not be considered valid due to fundamental issues in the way the test set was created.

As discussed in Section 3.3.2, the VBR dataset does not provide a predefined validation set, nor does its test set contain ground truth pose information. To work around this limitation, we constructed a test set using images from the training set. This approach introduced several issues that significantly impacted the validity of the experimental results.

The primary problems with our artificially created test set are:

1. Lack of Temporal, Weather, and Viewpoint Differences:
    - Since the test set was derived from the training set, there is minimal difference between the two in terms of lighting conditions, weather variations, and viewpoint changes. This eliminates many of the real-world challenges typically encountered in visual localization.
2. Use of Consecutive Image Sequences:
    - The test set consists of image sequences that follow a continuous trajectory where the operator passed through the same location at least twice. As a

result, one session of that trajectory is in the training set, while another is in the test set.

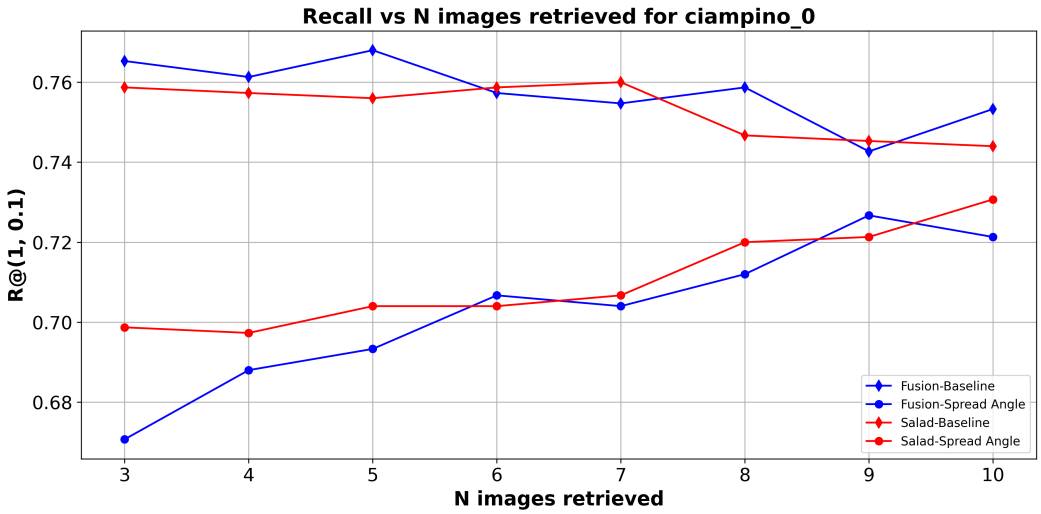
- This leads to two extreme scenarios:
  - In some cases, the test trajectory is nearly identical to the one in the training set, making image retrieval trivially easy.
  - In other cases, the test trajectory differs too much from the training set, making it impossible for the retrieval method to find relevant images.

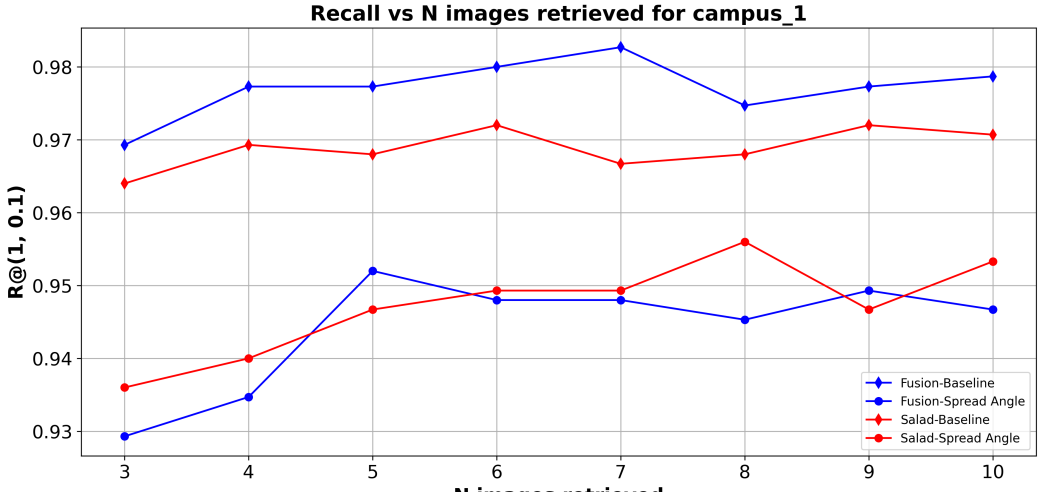
Due to these limitations, the results obtained in these experiments are unreliable. The artificially constructed test set fails to provide a meaningful evaluation of retrieval-based localization performance, as it either simplifies the task to an unrealistic degree or makes it completely unsolvable.

Despite these issues, we will report the experimental results in Table 4.6 and Figures 4.3, 4.4 for completeness, but we will refrain from interpreting or drawing conclusions from them. For more comprehensive result tables check chapter B of the appendix.

Dataset Phone	Method	Fusion		Salad	
		R@(1, 0.1)	R@(5, 1.0)	R@(1, 0.1)	R@(5, 1.0)
Ciampino	10 Images Retrieved	0.747	0.996	0.737	0.995
Ciampino	40 Images Retrieved	0.713	0.992	0.722	0.999
Ciampino	CL+LF 10 Images	0.747	0.996	0.737	0.995
Campus	10 Images Retrieved	0.977	1.0	0.968	1.0
Campus	40 Images Retrieved	0.949	1.0	0.975	1.0
Campus	CL+LF 10 Images	0.984	1.0	0.640	1.0
Colosseo	10 Images Retrieved	0.137	0.847	0.12	0.873
Colosseo	40 Images Retrieved	0.129	0.889	0.128	0.957
Colosseo	CL+LF 10 Images	0.157	0.824	0.125	0.873
Diag	10 Images Retrieved	0.552	0.932	0.565	0.931
Diag	40 Images Retrieved	0.551	0.92	0.58	0.928
Diag	CL+LF 10 Images	0.561	0.939	0.576	0.928
Pincio	10 Images Retrieved	0.184	0.679	0.16	0.635
Pincio	40 Images Retrieved	0.165	0.691	0.179	0.679
Pincio	CL+LF 10 Images	0.188	0.675	0.171	0.613
Spagna	10 Images Retrieved	0.239	0.688	0.177	0.723
Spagna	40 Images Retrieved	0.296	0.728	0.208	0.761
Spagna	CL+LF 10 Images	0.264	0.699	0.206	0.716

Table 4.6: Results of Experiment 1 and 2 on VBR.





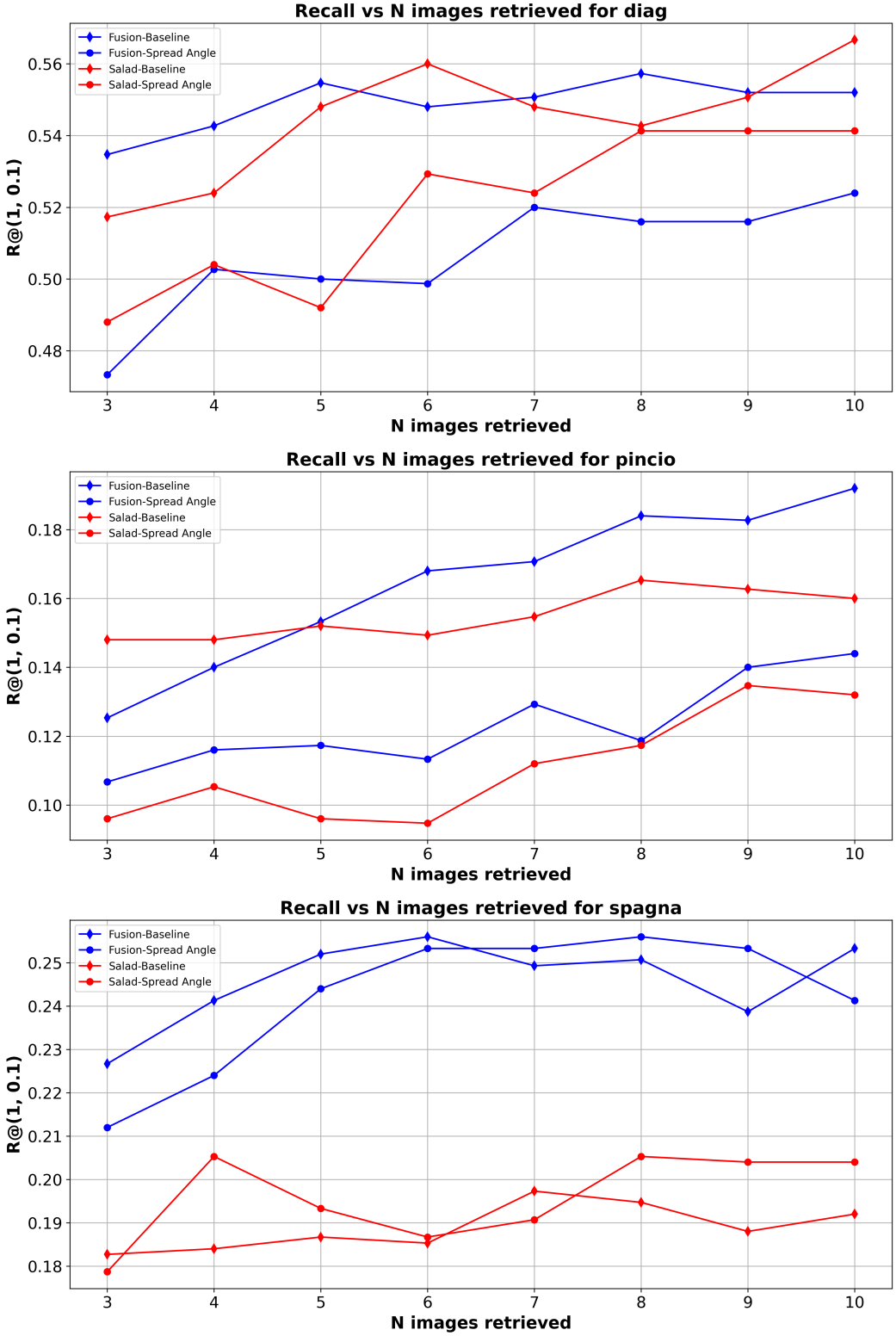


Figure 4.4: Results for Experiment 1 on VBR.

Chapter 5

Findings and Discussion

5.1 Role of Angular Differences in Retrieval Performance

The experimental results presented in Figure 4.1, 4.2 and Tables A.1, A.2, A.3 demonstrate that incorporating angular diversity in the image retrieval phase can improve pose estimation accuracy under certain conditions. However, the benefits of this approach vary across different scenes and depend on the specific global feature extractor used.

For the HGE scene with phone query images, which is characterized by a highly diverse mix of indoor and outdoor environments around a historical university building, the method that leverages a wider angular spread generally outperforms the baseline selection. In particular, when using the fusion-based global extractor, the performance difference between the baseline and the angle-spread approach can reach up to 6% recall for n=4 images. This indicates that, in the HGE scene with phone query images, maximizing angular diversity consistently enhances pose estimation, albeit with varying degrees of improvement depending on the number of images retrieved.

In the CAB scene with phone query images, captured within confined indoor spaces such as corridors and small rooms, the impact of angular spread differs between global extractors. When using the SALAD-based method, the angular spread strategy yields, on average, about a 1% improvement in recall, peaking at a 2.5% advantage for n=6 images. Conversely, for the fusion-based extractor in the CAB scene with phone query images, the angular spread approach appears to have minimal influence, with both baseline and angle-spread methods producing similar results regardless of the number of images retrieved. This can be attributed to the inherent difficulty of differentiating between visually similar images in such constrained environments.

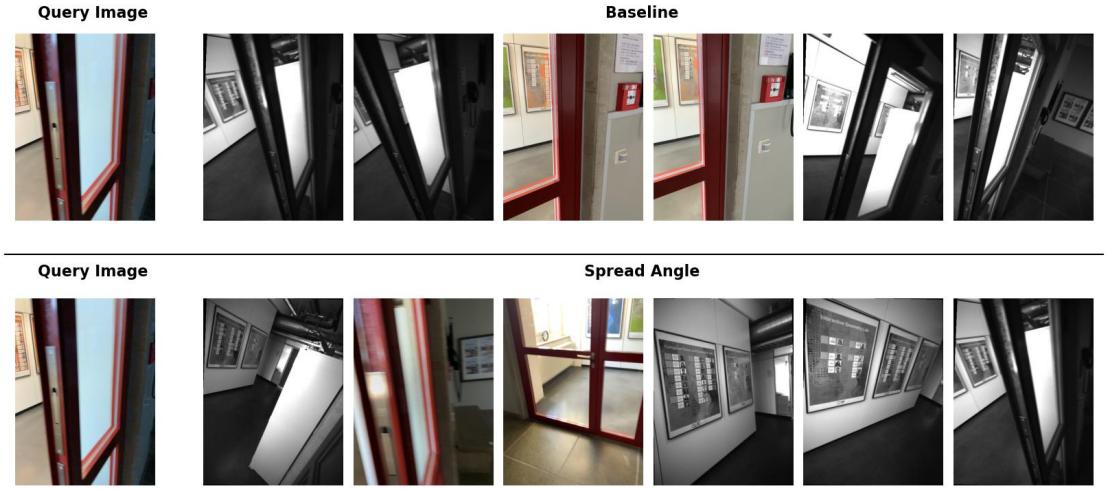


Figure 5.1: Comparison of retrieved images for the CAB scene with phone query images using SALAD. The query image is shown on the left, while the top row presents the first six images retrieved using the baseline method, and the bottom row shows the first six images retrieved using the spread angle method. As observed, the baseline method retrieves nearly identical images, primarily focused on the subject from a close distance, whereas the spread angle method retrieves more diverse images that depict both the subject and the surrounding environment from different angles.

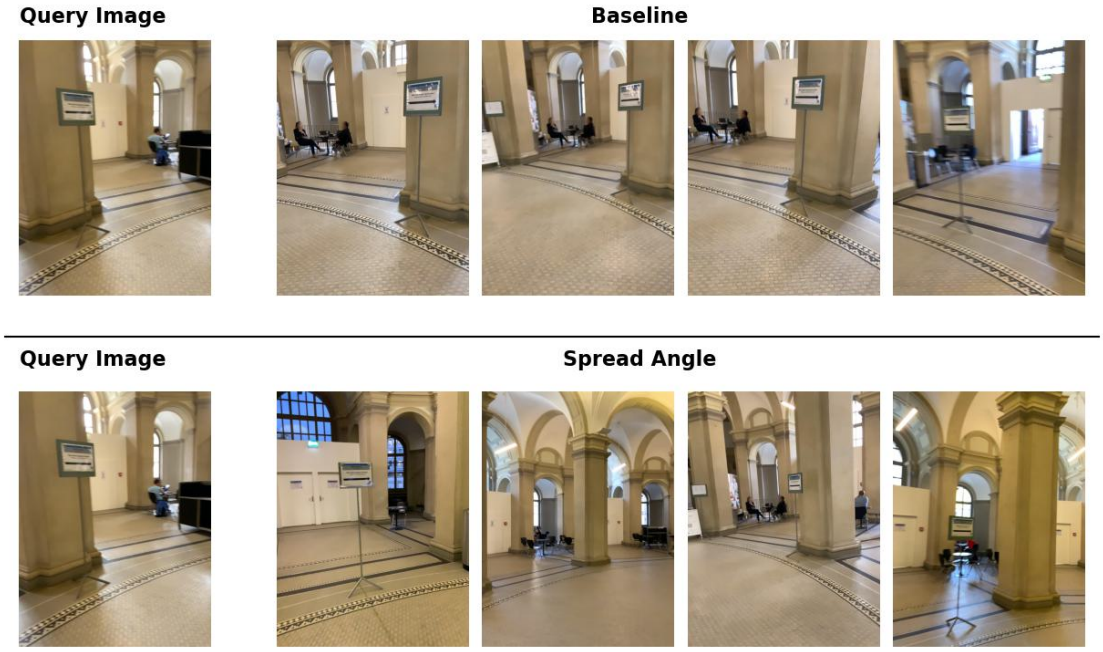


Figure 5.2: Comparison of retrieved images for the HGE scene with phone query images using fusion. The query image is displayed on the left, with the top row showing the first six images retrieved using the baseline method and the bottom row showing the first six images retrieved using the spread angle method. As observed, the baseline method retrieves nearly identical images taken from very similar viewpoints. In contrast, the spread angle method selects a more diverse set of images, capturing the subject and its surroundings from different perspectives.

For the LIN scene with phone query images, the effect of angular spread is generally non-influential. The recall scores for the baseline and angle-spread methods are very close, and in some cases, the baseline method even slightly outperforms the angle-spread selection, particularly when using the SALAD-based extractor.

Additionally, the two global extractors exhibit different sensitivities to angular diversity. For example, in the HGE scene with phone query images at n=4, the fusion-based extractor showed the most pronounced improvement when angular spread was applied, suggesting that its performance benefits more from diverse viewpoints compared to the SALAD-based extractor.

These results can be explained by the intrinsic characteristics of each scene. The HGE scene is notably diverse, with images capturing varied aspects of a historical university building, each room and outdoor area is distinct in scale and visual content, and repetitive patterns are minimal. This diversity allows the retrieval process to benefit significantly from a wide range of viewing angles. In contrast, the CAB scene consists of many similar indoor environments, such as corridors and staircases, where images from different locations can appear nearly identical. In such cases, global features struggle to differentiate between them, and the enforced angular spread may even lead to the selection of less relevant images.

Instead, for the HGE and LIN scenes using Hololens query images, we observe no appreciable difference between the baseline and angle-spread methods. In particular, when using the SALAD-based extractor in the HGE scene with a small number of retrieved images, the baseline approach even performs better. This suggests that for these specific settings, angular diversity does not contribute positively to retrieval performance. In the CAB scene with Hololens query images, we still observe that the angle-spread method provides a benefit when leveraging SALAD, but with the fusion-based extractor, there is no noticeable improvement.

It is logical to conclude that the benefit of incorporating angular spreadness in image retrieval is highly dependent on the nature of the dataset, the characteristics of the scene, and the capacity of the global feature extractor to represent images from different viewpoints consistently. In scenarios where the available database images lack sufficient angular diversity, efforts to maximize angular spread can be counterproductive. Conversely, in environments with rich viewpoint variation, enhancing angular diversity can significantly improve retrieval performance. Moreover, the different behaviours of the two used global extractors suggest that the effectiveness of angular spread depends not only on the scene characteristics but also on the choice of the global feature extractor. Different global extractors capture similarities and differences in images in distinct ways, depending on their strengths and feature representations. As a result, they can behave differently when enforcing maximum angular spread in the retrieved images, with some extractors benefiting from a wider viewpoint diversity while others may not effectively leverage this

additional variation. These findings underscore the importance of tailoring the retrieval strategy to the specific characteristics of the scene and the capabilities of the employed global feature extractor.

5.2 Comparative Analysis of Models

From all the experiments (Figures 4.1, 4.2 and Table 4.1, 4.2, 4.4, 4.5) conducted across different methods, scenes, and retrieval settings, the SALAD-based global feature extractor consistently outperformed the fusion approach combining NetVLAD and AP-GeM. This performance gap was evident regardless of the number of retrieved images and across both phone and Hololens query sets. On average, SALAD demonstrated a recall improvement of approximately 6-13% for R@(1,0.1), further confirming its superior ability to retrieve more relevant images for pose estimation.

These results highlight the advancements in global feature extraction techniques, with SALAD significantly surpassing the much older NetVLAD + AP-GeM combination. The performance gap can be attributed to the more refined and expressive representations learned by SALAD, which leverage modern self-supervised learning techniques and better capture scene structure and semantic information.

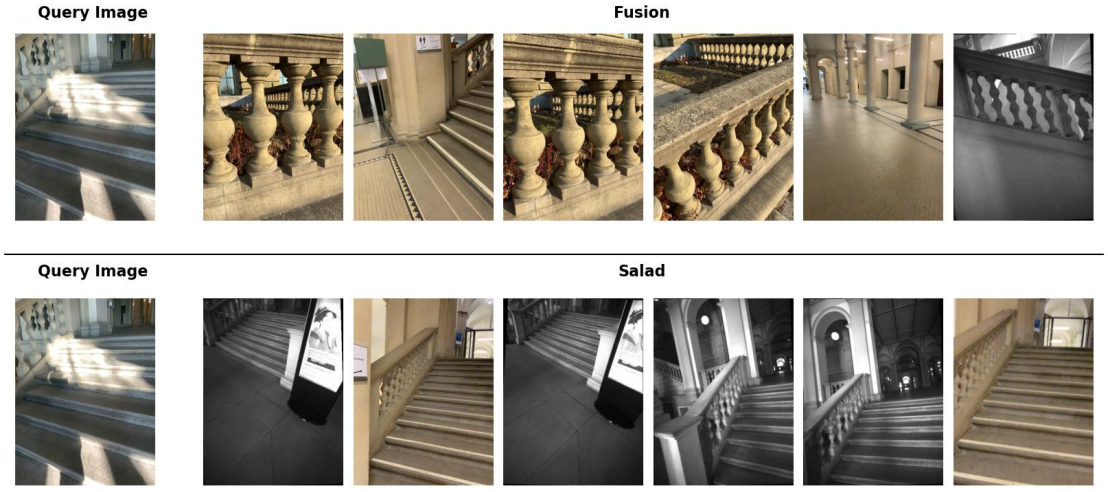


Figure 5.3: Comparison of HGE retrieved images with phone queries using different global extractors. The query is displayed on the left, with the top row showing the first six images retrieved using the fusion-based global extractor and the bottom row showing the first six images retrieved using the SALAD-based global extractor. In this example, SALAD retrieves more images that are relevant to the query, demonstrating its superior ability to capture pertinent visual information for image retrieval and pose estimation.

Moreover, the superior performance of SALAD was consistent across different retrieval strategies, including baseline retrieval (top 10 images) and expanded retrieval (top 40 images). This suggests that SALAD not only retrieves better-matching images but also improves robustness in diverse conditions, making it a more reliable choice for visual localization tasks.

While our results emphasize the importance of selecting a strong global feature extractor, an interesting direction for further improving performance would be to train or at least fine-tune a global feature extractor specifically for the task of visual localization. The requirements for an effective global feature extractor in this pipeline can differ significantly from those in other tasks, such as geolocalization, where broader scene-level information might be more relevant than viewpoint diversity. By fine-tuning a global extractor on visual localization-specific datasets, we could further optimize feature representations for this particular task, leading to even greater improvements in retrieval and pose estimation accuracy.

Additionally, training a model specifically for visual localization could help address the uncertainty observed in our first experiment regarding the impact of angular spreadness in retrieval. A global feature extractor adapted to this task could inherently learn the optimal degree of viewpoint diversity needed for different scene types, thereby resolving some of the ambiguities we observed when enforcing angular spread. This suggests that beyond simply swapping feature extractors, a task-specific adaptation could not only enhance retrieval accuracy but also provide more principled guidance on whether enforcing viewpoint diversity is beneficial in a given scenario.

The results also reinforce the importance of using state-of-the-art global descriptors for improving pose estimation pipelines. As seen in our experiments, an enhanced retrieval step directly translates to better localization accuracy, emphasizing the critical role of global feature extraction in visual place recognition. Given the significant improvements observed with SALAD, it is evident that modern retrieval models leveraging self-supervised learning and stronger feature representations can substantially enhance the performance of visual localization systems. Furthermore, fine-tuning these models for the specific needs of visual localization could unlock additional performance gains and help optimize retrieval strategies, particularly regarding viewpoint diversity.

5.3 Impact of Local Features

Our experimental results, presented in Tables 4.4 and 4.5, demonstrate that incorporating additional information, such as the known positions of database images for clustering and local feature-based filtering during retrieval, can enhance pose estimation performance. By integrating spatial priors and leveraging local

features earlier in the retrieval process, we observed notable improvements in recall, particularly for weaker global feature extractors.

The most significant improvements were seen with the fusion-based retriever, where the inclusion of these additional cues led to recall gains of up to 6% for the HGE scene with phone query images. This suggests that the NetVLAD + AP-GeM combination benefits substantially from supplementary spatial and geometric information, likely due to its more limited capacity to distinguish relevant images compared to more modern extractors like SALAD. By refining the retrieval process with local feature filtering and clustering, the fusion method was able to select more relevant images, which in turn improved the quality of the pose estimation pipeline.

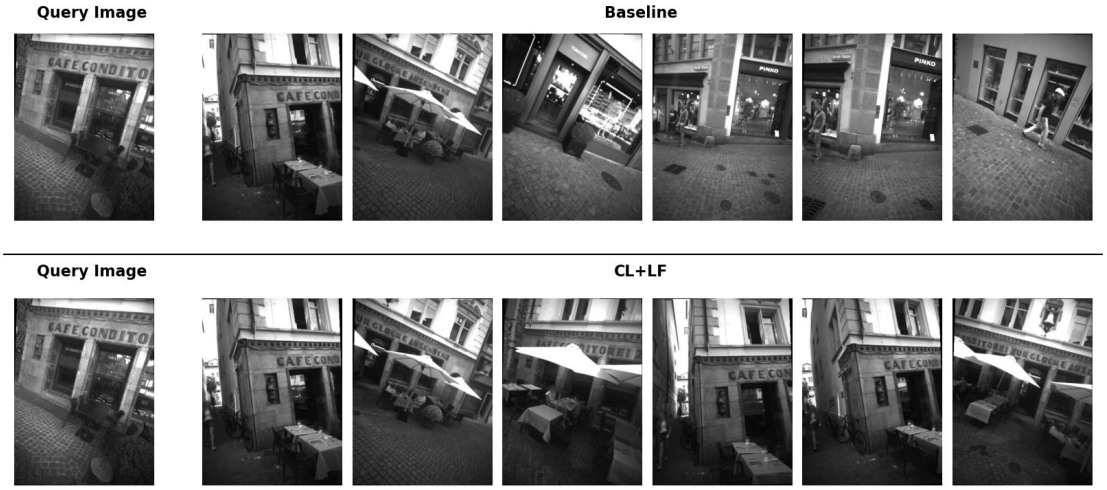


Figure 5.4: Comparison of retrieved images for the LIN scene with HoloLens query images using the baseline method versus our proposed clustering and local feature matching approach. The query image is shown on the left, with the top row displaying the first six images retrieved using the baseline method and the bottom row showing the first six images retrieved using the clustering + local feature matching approach. The additional information provided by clustering and leveraging local features allows for the retrieval of more relevant images, correctly depicting the target café and improving localization accuracy.

For the SALAD-based retriever, the impact of these modifications was more nuanced. While improvements of around 1-2% were observed in some cases, there were also instances where performance slightly decreased. This suggests that SALAD, being a stronger global feature extractor, may already retrieve highly relevant images, and further refining the selection based on clustering and local feature constraints might not always be beneficial. In some cases, enforcing these

additional retrieval constraints could potentially exclude useful images that would otherwise contribute positively to the final pose estimation.

These findings highlight the varying impact of local feature-based refinement depending on the strength of the global feature extractor. For weaker global extractors like the fusion method, the additional spatial and geometric cues provide crucial improvements, helping compensate for less discriminative global features. On the other hand, for more advanced methods like SALAD, the benefits are less pronounced and may even introduce trade-offs in certain scenarios.

Ultimately, these results emphasize the importance of balancing global and local feature information in the retrieval process. While spatial priors and local feature constraints can be highly beneficial, their effectiveness is influenced by the inherent capabilities of the global feature extractor, the scene characteristics, and the retrieval pipeline design.

5.4 Implications for Visual Localization Pipelines

The experiments presented in this work offer valuable insights into how different aspects of the image retrieval process influence the performance of visual localization pipelines. The findings suggest that improvements can be achieved by addressing three key areas: the choice of global feature extractor, the incorporation of additional local feature and positional information, and the enforcement of angular spreadness in the retrieved images.

1. Global Feature Extractor Comparison: Fusion vs. SALAD

Among all the experiments conducted, the comparison between the fusion-based approach (NetVLAD + AP-GeM) and the SALAD-based global feature extractor clearly shows that SALAD consistently outperforms fusion. This performance gap highlights the superiority of SALAD, which leverages modern self-supervised learning techniques and more expressive feature representations to capture scene structure and semantic content more effectively than the older fusion approach.

Notably, swapping from an off-the-shelf fusion extractor to an off-the-shelf SALAD extractor is one of the simplest modifications that can be made in the visual localization pipeline. This change is low effort but high reward, as it directly translates to significantly improved localization accuracy. However, it is important to note that in real-time applications where hardware resources and computational time are constrained, the increased computational load associated with SALAD must be carefully evaluated. The balance between improved accuracy and computational efficiency is crucial for practical deployments.

2. Leveraging Additional Local Information

Our third experiment demonstrated that incorporating additional information, specifically, the known positions of database images and local feature matching data, can further enhance the image retrieval process. This additional information is already available within typical visual localization pipelines and does not require computation from scratch; rather, it can be efficiently retrieved from memory. The experiment showed that lower-performance global extractors like the fusion method benefit more from leveraging this supplementary data, achieving significant recall improvements when clustering images by position and selecting the best cluster based on local feature matches.

This finding suggests that underutilizing available positional and local feature data represents a missed opportunity in improving the pipeline's performance. However, it should be noted that the additional computational cost associated with clustering and local feature matching can be significant. Therefore, the trade-off between improved recall and increased processing time must be carefully evaluated to ensure the overall efficiency of the system.

3. Impact of Angular Spreadness in Retrieval

The experiment investigating angular spreadness revealed that enforcing a wider diversity of viewpoints in the retrieved images can, in many cases, enhance pose estimation accuracy. In some datasets, maximizing angular diversity resulted in substantial improvements, while in others, its impact was minimal or even counterproductive. The effectiveness of angular spreadness depends on several factors, including the inherent structure and morphology of the dataset, the availability of diverse viewpoints for the same scene or subject, and the capabilities of the global and local feature extractors.

In some scenarios, such as the HGE scene with phone query images, increasing angular spread yielded notable improvements, up to a 6% gain in recall for the fusion-based extractor. In contrast, for certain scenes and with specific query types (e.g., Hololens queries in HGE and LIN), the baseline retrieval method performed comparably or even slightly better than the angular spread approach. These mixed results indicate that while angular spreadness has the potential to be beneficial, its effectiveness is highly context-dependent and warrants further investigation with more extensive experiments and carefully curated datasets.

Overall Implications

Collectively, these experiments underscore the importance of carefully balancing global and local feature information in visual localization pipelines. The evidence

strongly suggests that modern global feature extractors like SALAD can substantially enhance localization performance compared to older methods such as NetVLAD + AP-GeM. Additionally, leveraging available spatial and local feature data can further boost performance, particularly for pipelines that rely on less discriminative global features. Finally, while increasing the angular diversity of retrieved images holds promise, its benefits are highly dependent on various factors and must be tailored to the specific conditions of the application.

These findings indicate that even relatively straightforward modifications, such as replacing the global feature extractor or integrating additional retrieval cues, can lead to significant improvements in visual localization. However, practical deployment requires careful consideration of computational costs, especially in scenarios where real-time performance is critical. Future work should further explore these trade-offs and develop optimized strategies that balance accuracy and efficiency in diverse operational environments.

Chapter 6

Conclusion and Future Work

6.1 Summary of Contributions

This work investigated the impact of different retrieval strategies and global feature extractors on visual localization pipelines. Through a series of controlled experiments, we explored three key aspects: (1) the role of angular spreadness in retrieval performance, (2) the comparative effectiveness of different global feature extractors, particularly SALAD vs. the fusion-based approach, and (3) the impact of incorporating additional spatial and local feature information.

Our findings demonstrate that SALAD consistently outperforms the older fusion-based approach, making it a highly effective choice for improving image retrieval in localization pipelines. Additionally, we showed that leveraging already available positional and local feature data enhances performance, particularly for weaker global feature extractors. Finally, we found that enforcing angular spreadness in retrieved images can be beneficial in many cases, but its impact is scene-dependent and requires further investigation.

6.2 Limitations of the Current Work

While our experiments provide meaningful insights, certain limitations must be acknowledged. First, the dataset constraints may have influenced our findings. The experiments were conducted on a limited set of scenes, and the observed benefits of angular spreadness or additional retrieval cues might not generalize to all environments. Larger and more diverse datasets could provide a clearer picture of these effects.

Second, computational efficiency remains a concern. While SALAD offers superior retrieval performance, its increased computational cost may pose challenges for real-time localization applications. Similarly, incorporating additional local feature

matching and spatial clustering improves recall but also introduces processing overhead. The trade-off between accuracy and efficiency needs further evaluation, particularly for real-world applications with strict latency constraints.

Lastly, the role of angular spreadness remains somewhat ambiguous. While we observed improvements in some scenarios, the factors that determine its effectiveness, such as dataset structure, viewpoint availability, and feature extractor choice, require more systematic exploration. resources, or specific model issues.

6.3 Directions for Future Research

Several avenues for future research emerge from this work. One key direction is expanding our experiments to larger and more diverse datasets, covering a broader range of environments, including outdoor urban spaces, complex indoor settings, and dynamic scenes with moving objects. Such datasets would help validate our findings and refine strategies for integrating retrieval improvements into visual localization pipelines.

Another important area is exploring alternative global feature extractors and retrieval techniques. While SALAD outperformed the fusion approach, future work could examine even more advanced self-supervised or transformer-based models to further enhance retrieval performance.

Finally, future research should investigate how these retrieval improvements translate to downstream geolocalization tasks. While our work focused on retrieval performance, integrating these enhancements into full pose estimation pipelines could provide a more comprehensive assessment of their practical impact. Additionally, optimizing retrieval strategies for real-time applications by balancing accuracy and computational cost will be crucial for deploying these techniques in practical scenarios such as augmented reality navigation and autonomous systems.

By addressing these challenges, future work can build upon our findings to further refine and optimize visual localization pipelines, improving both their accuracy and efficiency in diverse environments.

Appendix A

Complete Results of Experiment 1 on Lamar

In this chapter, we present the complete results of Experiment 1, which investigates the impact of angular spreadness on image retrieval and pose estimation accuracy. The details of this experiment are discussed in Chapter 4.3.

Each of the following tables corresponds to a specific scene in the Lamar dataset and provides a comprehensive breakdown of the experimental results for angular spreadness. The tables report two key recall metrics:

- R@(1, 0.1): Recall at 1 image within a 0.1m threshold.
- R@(5, 1.0): Recall at 5 images within a 1.0m threshold.

The results are presented for different numbers of retrieved images, ranging from 3 to 10. We compare the performance of two global feature extractors:

- Fusion-based extractor (NetVLAD + AP-GeM)
- SALAD-based extractor (DINOv2 SALAD)

For each global feature extractor, we report results under both the baseline selection method and the angular spread selection method. The best result for each number of retrieved images is highlighted in red to emphasize the most effective approach in each case.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.515	0.684	0.505	0.657	0.510	0.702	0.525	0.712
4	0.523	0.687	0.520	0.674	0.533	0.715	0.525	0.715
5	0.530	0.684	0.538	0.689	0.533	0.715	0.543	0.717
6	0.543	0.684	0.528	0.689	0.535	0.720	0.561	0.720
7	0.548	0.687	0.538	0.689	0.566	0.728	0.575	0.720
8	0.556	0.694	0.550	0.689	0.563	0.730	0.571	0.717
9	0.553	0.689	0.548	0.692	0.556	0.725	0.568	0.730
10	0.553	0.697	0.548	0.689	0.561	0.732	0.573	0.730

Table A.1: Experimental results for CAB scene for phone queries.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.552	0.817	0.607	0.857	0.624	0.884	0.592	0.868
4	0.586	0.838	0.651	0.870	0.641	0.891	0.634	0.884
5	0.590	0.844	0.639	0.870	0.647	0.891	0.666	0.895
6	0.624	0.859	0.658	0.872	0.641	0.893	0.666	0.895
7	0.643	0.861	0.660	0.872	0.653	0.895	0.672	0.895
8	0.645	0.870	0.655	0.874	0.668	0.899	0.672	0.897
9	0.643	0.863	0.668	0.876	0.670	0.897	0.689	0.901
10	0.641	0.874	0.672	0.876	0.681	0.901	0.683	0.899

Table A.2: Experimental results for HGE scene for phone queries.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.777	0.909	0.777	0.910	0.800	0.937	0.757	0.947
4	0.777	0.911	0.780	0.912	0.797	0.944	0.795	0.952
5	0.797	0.917	0.792	0.926	0.818	0.947	0.803	0.960
6	0.795	0.916	0.803	0.927	0.815	0.947	0.807	0.957
7	0.797	0.919	0.805	0.924	0.821	0.949	0.815	0.959
8	0.798	0.919	0.800	0.924	0.826	0.952	0.821	0.957
9	0.800	0.922	0.803	0.924	0.825	0.950	0.837	0.960
10	0.798	0.921	0.798	0.926	0.835	0.954	0.836	0.960

Table A.3: Experimental results for LIN scene for phone queries.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.417	0.581	0.404	0.578	0.465	0.689	0.456	0.680
4	0.442	0.603	0.434	0.605	0.484	0.694	0.491	0.706
5	0.457	0.613	0.453	0.621	0.496	0.711	0.512	0.716
6	0.468	0.630	0.465	0.632	0.507	0.712	0.518	0.727
7	0.463	0.638	0.471	0.640	0.517	0.717	0.526	0.733
8	0.483	0.647	0.481	0.648	0.515	0.731	0.539	0.742
9	0.482	0.655	0.483	0.649	0.536	0.734	0.545	0.747
10	0.488	0.650	0.484	0.656	0.537	0.736	0.543	0.743

Table A.4: Experimental results for CAB scene for hololens queries.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.425	0.624	0.391	0.637	0.543	0.755	0.496	0.735
4	0.447	0.642	0.432	0.647	0.564	0.770	0.531	0.750
5	0.452	0.647	0.452	0.657	0.583	0.779	0.558	0.762
6	0.452	0.656	0.455	0.661	0.587	0.778	0.569	0.772
7	0.470	0.661	0.463	0.669	0.594	0.775	0.578	0.770
8	0.469	0.664	0.468	0.668	0.585	0.785	0.585	0.774
9	0.472	0.668	0.469	0.671	0.591	0.783	0.594	0.778
10	0.473	0.670	0.478	0.674	0.606	0.784	0.594	0.780

Table A.5: Experimental results for HGE scene for hololens queries.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.610	0.737	0.585	0.723	0.688	0.887	0.670	0.866
4	0.624	0.752	0.617	0.737	0.723	0.899	0.707	0.877
5	0.637	0.757	0.635	0.746	0.742	0.902	0.744	0.897
6	0.647	0.762	0.645	0.750	0.741	0.903	0.755	0.901
7	0.651	0.762	0.649	0.752	0.759	0.906	0.763	0.905
8	0.654	0.766	0.658	0.755	0.757	0.909	0.772	0.909
9	0.655	0.765	0.661	0.759	0.762	0.915	0.782	0.910
10	0.660	0.766	0.664	0.759	0.768	0.913	0.784	0.908

Table A.6: Experimental results for LIN scene for hololens queries.

Appendix B

Complete Results of Experiment 1 on VBR

In this chapter, we present the complete results of Experiment 1, which examines the impact of angular spreadness on image retrieval and pose estimation accuracy. The details of this experiment are discussed in Chapter 4.6.

Each of the following tables corresponds to a specific scene in the VBR dataset and provides a detailed breakdown of the experimental results for angular spreadness. The tables report two key recall metrics:

- R@(1, 0.1): Recall at 1 image within a 0.1m threshold.
- R@(5, 1.0): Recall at 5 images within a 1.0m threshold.

The results are shown for different numbers of retrieved images, ranging from 3 to 10. We compare the performance of two global feature extractors:

- Fusion-based extractor (NetVLAD + AP-GeM)
- SALAD-based extractor (DINOv2 SALAD)

For each global feature extractor, we report results under both the baseline selection method and the angular spread selection method.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.765	0.960	0.671	0.960	0.759	0.949	0.699	0.949
4	0.761	0.960	0.688	0.960	0.757	0.949	0.697	0.949
5	0.768	0.960	0.693	0.960	0.756	0.949	0.704	0.949
6	0.757	0.960	0.707	0.960	0.759	0.949	0.704	0.949
7	0.755	0.960	0.704	0.960	0.760	0.949	0.707	0.949
8	0.759	0.960	0.712	0.960	0.747	0.949	0.720	0.949
9	0.743	0.960	0.727	0.960	0.745	0.949	0.721	0.949
10	0.753	0.960	0.721	0.960	0.744	0.949	0.730	0.949

Table B.1: Experimental results for different retrieval methods on the Ciampino dataset.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.9693	1.000	0.9293	1.000	0.9640	1.000	0.9360	1.000
4	0.9773	1.000	0.9347	1.000	0.9693	1.000	0.9400	1.000
5	0.9773	1.000	0.9520	1.000	0.9680	1.000	0.9467	1.000
6	0.9800	1.000	0.9480	1.000	0.9720	1.000	0.9493	1.000
7	0.9827	1.000	0.9480	1.000	0.9667	1.000	0.9493	1.000
8	0.9747	1.000	0.9453	1.000	0.9680	1.000	0.9560	1.000
9	0.9773	1.000	0.9493	1.000	0.9720	1.000	0.9467	1.000
10	0.9787	1.000	0.9467	1.000	0.9707	1.000	0.9533	1.000

Table B.2: Experimental results for different retrieval methods on the Campus_1 dataset.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.1347	0.7720	0.1267	0.7627	0.1467	0.8040	0.1173	0.7920
4	0.1240	0.7813	0.1000	0.7747	0.1453	0.8147	0.1187	0.7960
5	0.1293	0.7933	0.0960	0.7853	0.1307	0.8173	0.1107	0.8040
6	0.1307	0.7893	0.1120	0.7907	0.1307	0.8147	0.1093	0.8253
7	0.1520	0.7947	0.1053	0.7893	0.1360	0.8173	0.1120	0.8160
8	0.1440	0.7973	0.1120	0.8013	0.1280	0.8213	0.1267	0.8240
9	0.1427	0.7987	0.1187	0.7987	0.1280	0.8227	0.1253	0.8293
10	0.1493	0.8027	0.1333	0.7960	0.1213	0.8213	0.1293	0.8267

Table B.3: Experimental results for different retrieval methods on the Colosseo dataset.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.5347	0.9133	0.4733	0.8973	0.5173	0.9147	0.4880	0.8880
4	0.5427	0.9227	0.5027	0.9107	0.5240	0.9173	0.5040	0.8987
5	0.5547	0.9320	0.5000	0.9080	0.5480	0.9280	0.4920	0.9067
6	0.5480	0.9320	0.4987	0.9093	0.5600	0.9320	0.5293	0.9173
7	0.5507	0.9293	0.5200	0.9173	0.5480	0.9320	0.5240	0.9120
8	0.5573	0.9360	0.5160	0.9293	0.5427	0.9293	0.5413	0.9240
9	0.5520	0.9293	0.5160	0.9267	0.5507	0.9280	0.5413	0.9240
10	0.5520	0.9347	0.5240	0.9253	0.5667	0.9307	0.5413	0.9253

Table B.4: Experimental results for different retrieval methods on the diag dataset.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.1253	0.6387	0.1067	0.5907	0.1480	0.6453	0.0960	0.5680
4	0.1400	0.6533	0.1160	0.6213	0.1480	0.6587	0.1053	0.6013
5	0.1533	0.6733	0.1173	0.6440	0.1520	0.6627	0.0960	0.6053
6	0.1680	0.6707	0.1133	0.6600	0.1493	0.6533	0.0947	0.6200
7	0.1707	0.6747	0.1293	0.6533	0.1547	0.6467	0.1120	0.6253
8	0.1840	0.6907	0.1187	0.6627	0.1653	0.6453	0.1173	0.6293
9	0.1827	0.6813	0.1400	0.6707	0.1627	0.6507	0.1347	0.6440
10	0.1920	0.6867	0.1440	0.6667	0.1600	0.6387	0.1320	0.6387

Table B.5: Experimental results for different retrieval methods on the Pincio dataset.

N	Fusion Baseline		Fusion Spread		Salad Baseline		Salad Spread	
	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)	R@(1,0.1)	R@(5,1)
3	0.2267	0.6613	0.2120	0.6547	0.1827	0.6360	0.1787	0.6267
4	0.2413	0.6653	0.2240	0.6560	0.1840	0.6400	0.2053	0.6307
5	0.2520	0.6560	0.2440	0.6560	0.1867	0.6387	0.1933	0.6227
6	0.2560	0.6533	0.2533	0.6613	0.1853	0.6333	0.1867	0.6320
7	0.2493	0.6547	0.2533	0.6600	0.1973	0.6373	0.1907	0.6373
8	0.2507	0.6587	0.2560	0.6587	0.1947	0.6360	0.2053	0.6347
9	0.2387	0.6613	0.2533	0.6600	0.1880	0.6387	0.2040	0.6307
10	0.2533	0.6547	0.2413	0.6627	0.1920	0.6373	0.2040	0.6373

Table B.6: Experimental results for different retrieval methods on the Spagna dataset.

Bibliography

[1] No'e Pion, M. Humenberger, Gabriela Csurka, Yohann Cabon, and Torsten Sattler. «Benchmarking Image Retrieval for Visual Localization». In: *2020 International Conference on 3D Vision (3DV)* (2020), pp. 483–494. URL: https://api.semanticscholar.org/CorpusID:227151822 (cit. on pp. 3, 11, 13, 14).

[2] Martin Humenberger, Yohann Cabon, Noé Pion, Philippe Weinzaepfel, Donghwan Lee, Nicolas Guérin, Torsten Sattler, and Gabriela Csurka. «Investigating the Role of Image Retrieval for Visual Localization: An Exhaustive Benchmark». In: *International Journal of Computer Vision* 130.7 (May 2022), pp. 1811–1836. ISSN: 1573-1405. DOI: 10.1007/s11263-022-01615-7. URL: http://dx.doi.org/10.1007/s11263-022-01615-7 (cit. on pp. 3, 13, 14).

[3] Martin Humenberger. *Invited talk, 3rd 3D-Deep Learning for Autonomous Driving (3D-DLAD) workshop*. July 2021 (cit. on pp. 5, 6, 11).

[4] T. Sattler, B. Leibe, and L. Kobbelt. «Fast Image-Based Localization using Direct 2D-to-3D Matching». In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2011 (cit. on p. 8).

[5] Pierre Moulon, Pascal Monasse, and Renaud Marlet. «Global Fusion of Relative Motions for Robust, Accurate and Scalable Structure from Motion». In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2013 (cit. on p. 8).

[6] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. «Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 39.9 (2017), pp. 1744–1756 (cit. on p. 8).

[7] Liu Liu, Hongdong Li, and Yuchao Dai. «Efficient Global 2D-3D Matching for Camera Localization in a Large-Scale 3D Map». In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2017 (cit. on p. 8).

[8] H. Taira, M. Okutomi, T. Sattler, M. Cimpoi, M. Pollefeys, J. Sivic, T. Pajdla, and A. Torii. «InLoc: Indoor Visual Localization with Dense Matching and View Synthesis». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* (2019) (cit. on p. 8).

[9] Johannes Lutz Schönberger and Jan-Michael Frahm. «Structure-from-Motion Revisited». In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016 (cit. on p. 8).

[10] Gabriela Csurka, Christopher R. Dance, and Martin Humenberger. *From Handcrafted to Deep Local Invariant Features*. arXiv:1807.10254. 2018 (cit. on p. 8).

[11] Jerome Revaud, Philippe Weinzaepfel, Cesar de Souza, and Martin Humenberger. «R2D2: Reliable and Repeatable Detectors and Descriptors». In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2019 (cit. on p. 8).

[12] David G. Lowe. «Distinctive Image Features from Scale-Invariant Keypoints». In: *International Journal of Computer Vision (IJCV)* 60.2 (2004), pp. 91–110 (cit. on p. 8).

[13] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. «D2-Net: A Trainable CNN for Joint Description and Detection of Local Features». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019 (cit. on p. 8).

[14] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. *SuperGlue: Learning Feature Matching with Graph Neural Networks*. 2020. arXiv: 1911.11763 [cs.CV]. URL: https://arxiv.org/abs/1911.11763 (cit. on pp. 8, 27).

[15] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. «SuperPoint: Self-Supervised Interest Point Detection and Description». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2018 (cit. on pp. 8, 25, 26).

[16] Johannes Lutz Schonberger, Hans Hardmeier, Torsten Sattler, and Marc Pollefeys. «Comparative Evaluation of Hand-Crafted and Learned Local Features». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 (cit. on p. 8).

[17] Laurent Kneip, Davide Scaramuzza, and Roland Siegwart. «A Novel Parametrization of the Perspective-Three-Point Problem for a Direct Computation of Absolute Camera Position and Orientation». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2011 (cit. on p. 8).

[18] M. Fischler and R. Bolles. «Random Sampling Consensus: A Paradigm for Model Fitting with Application to Image Analysis and Automated Cartography». In: *Communications of the ACM* 24 (1981), pp. 381–395 (cit. on p. 8).

[19] O. Chum and J. Matas. «Optimal Randomized RANSAC». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 30.8 (2008), pp. 1472–1482 (cit. on p. 8).

[20] Torsten Sattler et al. «Benchmarking 6DoF Outdoor Visual Localization in Changing Conditions». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018 (cit. on p. 9).

[21] Martin Humenberger et al. *Robust Image Retrieval-based Visual Localization using Kapture*. arXiv:2007.13867. 2020 (cit. on p. 9).

[22] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. *From Coarse to Fine: Robust Hierarchical Localization at Large Scale*. 2019. arXiv: `1812.03506 [cs.CV]`. URL: `https://arxiv.org/abs/1812.03506` (cit. on p. 9).

[23] Eric Brachmann and Carsten Rother. «Learning Less is More – 6D Camera Localization via 3D Surface Regression». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018 (cit. on p. 9).

[24] Eric Brachmann and Carsten Rother. «Expert Sample Consensus Applied to Camera Re-localization». In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2019 (cit. on p. 9).

[25] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. «Hierarchical Scene Coordinate Classification and Regression for Visual Localization». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020 (cit. on p. 9).

[26] Eric Brachmann and Carsten Rother. *Visual Camera Re-localization from RGB and RGB-D Images using DSAC*. arXiv preprint. 2020 (cit. on p. 9).

[27] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. «Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2013 (cit. on p. 9).

[28] Carlo Masone and Barbara Caputo. «A Survey on Deep Visual Place Recognition». In: *IEEE Access* 9 (2021), pp. 19516–19547 (cit. on p. 10).

[29] Relja Arandjelović, Petr Gronát, Akihiko Torii, Tomás Pajdla, and Josef Sivic. «NetVLAD: CNN Architecture for Weakly Supervised Place Recognition». In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), pp. 5297–5307. URL: https://api.semanticscholar.org/CorpusID:44604205 (cit. on pp. 10, 20).

[30] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. «24/7 Place Recognition by View Synthesis». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015 (cit. on p. 10).

[31] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. «End-to-End Learning of Deep Visual Representations for Image Retrieval». In: *International Journal of Computer Vision (IJCV)* 124 (2017), pp. 237–254 (cit. on p. 10).

[32] Alex Kendall and Roberto Cipolla. «Geometric Loss Functions for Camera Pose Regression with Deep Learning». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 (cit. on pp. 10, 12).

[33] Jerome Revaud, Jon Almazan, Rafael Sampaio de Rezende, and Cesar Roberto de Souza. «Learning with Average Precision: Training Image Retrieval with a Listwise Loss». In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2019 (cit. on p. 10).

[34] Akihiko Torii, Josef Sivic, and Tomas Pajdla. «Visual Localization by Linear Combination of Image Descriptors». In: *Proceedings of the International Conference on Computer Vision Workshops (ICCV Workshops)*. 2011 (cit. on p. 10).

[35] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. «Understanding the Limitations of CNN-based Absolute Camera Pose Regression». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019 (cit. on p. 10).

[36] Wei Zhang and Jana Kosecka. «Image Based Localization in Urban Environments». In: *Proceedings of the International Conference on 3D Vision, Processing and Transmission (3DPVT)*. 2006 (cit. on p. 10).

[37] Qunjie Zhou, Torsten Sattler, Marc Pollefeys, and Laura Leal-Taixe. «To Learn or not to Learn: Visual Localization from Essential Matrices». In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2020 (cit. on p. 10).

[38] Alex Kendall, Matthew Grimes, and Roberto Cipolla. «PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization». In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015 (cit. on p. 11).

[39] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. «Geometry Aware Learning of Maps for Camera Localization». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018 (cit. on p. 12).

[40] Abhinav Valada, Noha Radwan, and Wolfram Burgard. «Deep Auxiliary Learning for Visual Localization and Odometry». In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2018 (cit. on p. 12).

[41] N. Radwan, A. Valada, and Wolfram Burgard. «Vlocnet++: Deep Multitask Learning for Semantic Visual Localization and Odometry». In: *Robotics and Automation Letters (RA-L)* 3.4 (2018), pp. 4407–4414 (cit. on p. 12).

[42] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. «DSAC – Differentiable RANSAC for Camera Localization». In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017 (cit. on p. 12).

[43] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. «SANet: Scene Agnostic Network for Camera Localization». In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2019 (cit. on p. 12).

[44] David G. Lowe. «Method and Apparatus for Identifying Scale Invariant Features in an Image and Use of Same for Locating an Object in an Image». US6005548A. Dec. 1999. URL: `https://patents.google.com/patent/US6005548A/en` (cit. on pp. 14, 17, 20, 25).

[45] Paul-Edouard Sarlin, Mihai Dusmanu, Johannes L. Schönberger, Pablo Speciale, Lukas Gruber, Viktor Larsson, Ondrej Miksik, and Marc Pollefeys. «LaMAR: Benchmarking Localization and Mapping for Augmented Reality». In: *ECCV*. 2022 (cit. on pp. 16, 17, 29, 31).

[46] Leonardo Brizi, Emanuele Giacomini, Luca Di Giammarino, Simone Ferrari Omar Salem, Lorenzo De Rebotti, and Giorgio Grisetti. *VBR: A Vision Benchmark in Rome*. 2024 (cit. on pp. 17, 31, 32).

[47] Laurent Kneip. «Real-Time Scalable Structure from Motion: From Fundamental Geometric Vision to Collaborative Mapping». PhD thesis. Jan. 2013 (cit. on p. 17).

[48] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. «ORB: An efficient alternative to SIFT or SURF». In: *2011 International Conference on Computer Vision*. 2011, pp. 2564–2571. DOI: 10.1109/ICCV.2011.6126544 (cit. on pp. 18, 25, 27).

[49] Feng Lu, Xinyao Zhang, Canming Ye, Shuting Dong, Lijun Zhang, Xiangyuan Lan, and Chun Yuan. «SuperVLAD: Compact and Robust Image Descriptors for Visual Place Recognition». In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. URL: https://openreview.net/forum?id=bZpZMdY1sj (cit. on p. 21).

[50] Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias Fischer. *Patch-NetVLAD: Multi-Scale Fusion of Locally-Global Descriptors for Place Recognition*. 2021. arXiv: 2103.01486 [cs.CV]. URL: https://arxiv.org/abs/2103.01486 (cit. on p. 21).

[51] Anuradha Uggi and Sumohana S. Channappayya. «MS-NetVLAD: Multi-Scale NetVLAD for Visual Place Recognition». In: *IEEE Signal Processing Letters* 31 (2024), pp. 1855–1859. DOI: 10.1109/LSP.2024.3425279 (cit. on p. 21).

[52] Gabriele Berton, Carlo Masone, and Barbara Caputo. *Rethinking Visual Geo-localization for Large-Scale Applications*. 2022. arXiv: 2204.02287 [cs.CV]. URL: https://arxiv.org/abs/2204.02287 (cit. on pp. 21, 22).

[53] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. *MixVPR: Feature Mixing for Visual Place Recognition*. 2023. arXiv: 2303.02190 [cs.CV]. URL: https://arxiv.org/abs/2303.02190 (cit. on pp. 21, 22).

[54] Bingyi Cao, Andre Araujo, and Jack Sim. *Unifying Deep Local and Global Features for Image Search*. 2020. arXiv: 2001.05027 [cs.CV]. URL: https://arxiv.org/abs/2001.05027 (cit. on p. 21).

[55] Akihiko Torii, Relja Arandjelović, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. «24/7 place recognition by view synthesis». In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1808–1817 (cit. on pp. 21, 22).

[56] Filip Radenovic, Giorgos Tolias, and Ondrej Chum. «Fine-Tuning CNN Image Retrieval with No Human Annotation». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 41.7 (2019), pp. 1655–1668 (cit. on pp. 21, 22).

[57] Jerome Revaud, Jon Almazan, Rafael S. Rezende, and Cesar Roberto de Souza. «Learning With Average Precision: Training Image Retrieval With a Listwise Loss». In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019 (cit. on p. 22).

[58] Marc'Aurelio Ranzato, Fu Jie Huang, Y-Lan Boureau, and Yann LeCun. «Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition». In: *2007 IEEE Conference on Computer Vision and Pattern Recognition*. 2007, pp. 1–8 (cit. on p. 22).

[59] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. *Particular object retrieval with integral max-pooling of CNN activations*. 2016. arXiv: 1511.05879 [cs.CV]. URL: https://arxiv.org/abs/1511.05879 (cit. on p. 22).

[60] Artem Babenko and Victor Lempitsky. *Aggregating Deep Convolutional Features for Image Retrieval*. 2015. arXiv: 1510.07493 [cs.CV]. URL: https://arxiv.org/abs/1510.07493 (cit. on p. 22).

[61] Sergio Izquierdo and Javier Civera. «Optimal Transport Aggregation for Visual Place Recognition». In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2024 (cit. on pp. 23, 24).

[62] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. «SURF: Speeded Up Robust Features». In: *Computer Vision – ECCV 2006*. Ed. by Aleš Leonardis, Horst Bischof, and Axel Pinz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417 (cit. on p. 25).

[63] E. Rosten and T. Drummond. «Fusing points and lines for high performance tracking». In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2. 2005, 1508–1515 Vol. 2 (cit. on p. 27).

[64] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. *LightGlue: Local Feature Matching at Light Speed*. 2023. arXiv: 2306.13643 [cs.CV]. URL: https://arxiv.org/abs/2306.13643 (cit. on pp. 27, 28).

[65] Marius Muja and David G. Lowe. «Scalable Nearest Neighbor Algorithms for High Dimensional Data». In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.11 (2014), pp. 2227–2240 (cit. on p. 27).

[66] Martin A. Fischler and Robert C. Bolles. «Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography». In: *Commun. ACM* 24.6 (June 1981), pp. 381–395. ISSN: 0001-0782. URL: https://doi.org/10.1145/358669.358692 (cit. on p. 27).

[67] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. *LoFTR: Detector-Free Local Feature Matching with Transformers*. 2021. arXiv: 2104.00680 [cs.CV]. URL: https://arxiv.org/abs/2104.00680 (cit. on p. 27).

[68] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. *COTR: Correspondence Transformer for Matching Across Images*. 2021. arXiv: 2103.14167 [cs.CV]. URL: https://arxiv.org/abs/2103.14167 (cit. on p. 27).