NANYANG TECHNOLOGICAL UNIVERSITY

STUDY OF LOCAL DESCRIPTORS FOR IMPROVED VISUAL PLACE RECOGNITION

Hu Tianrun

School of Computer Science and Engineering

2024

NANYANG TECHNOLOGICAL UNIVERSITY

SCSE8338

STUDY OF LOCAL DESCRIPTORS FOR IMPROVED VISUAL PLACE RECOGNITION

Submitted in Partial Fulfilment of the Requirements
for the Degree of Bachelor of Engineering in Computer Science
of the Nanyang Technological University

by

Hu Tianrun

School of Computer Science and Engineering
2024

Abstract

This research addresses a crucial challenge in robotics: enhancing Visual Place Recognition (VPR) within Simultaneous Localization and Mapping (SLAM). VPR is essential for robotic navigation and mapping, but it struggles to balance global descriptors' speed with local descriptors' precision. Traditional methods use global descriptors for initial candidate generation, followed by refinement with local descriptors to ensure accuracy. However, this re-ranking process significantly slows down processing, posing a challenge for real-time applications.

This project aims to develop a method that accelerates the local descriptor re-ranking process while maintaining accuracy. Overcoming this challenge is key to advancing VPR techniques and enabling more efficient and responsive robotic operations. Our approach involves adopting advanced algorithms and optimisation strategies to streamline the processing of local descriptors.

Building on previous work on attentive patches, we identify limitations and propose slight improvements to enhance the efficiency of VPR within SLAM. Our results suggest a promising direction for more robust and agile robotic navigation and mapping capabilities. This study contributes to the theoretical understanding of local descriptor matching and offers a practical solution to a longstanding efficiency challenge.

Acknowledgments

I extend my deepest gratitude to a number of individuals whose support was invaluable in the completion of this project.

First and foremost, I wish to express my sincere appreciation to Prof. Lam, whose expertise, guidance, and unwavering support have been pivotal throughout this research journey. Prof. Lam's insights and encouragement not only shaped this project but also inspired me to pursue excellence in my work.

I am also profoundly thankful to my family, whose love, understanding, and endless encouragement have been my cornerstone. Their belief in my abilities and their unwavering support have been my constant source of motivation and strength.

My heartfelt thanks go to my friends, who have been there for me through the highs and lows of this journey. Their companionship, reassurance, and constructive feedback have been invaluable, making this challenging journey a memorable and enjoyable experience.

I would also like to extend my gratitude to the senior members and PhD candidates in the lab, especially Dongshuo Zhang. Dongshuo's generosity in sharing his reference code and his willingness to offer advice and insights have significantly contributed to the success of this project. His assistance was a beacon of support that helped navigate the complexities of this research.

This project would not have been possible without the collective support, guidance, and encouragement of all those mentioned. I am deeply thankful to each one of you for your invaluable contribution to my journey.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

The burgeoning field of robotics has witnessed significant advancements in autonomously navigating and mapping environments, with Visual Place Recognition (VPR) emerging as a cornerstone. VPR, as the critical aspect of the Simultaneous Localisation and Mapping (SLAM) [1], enables robots to recognise previously visited locations and correct the accumulated drifting error in mapping generation. At the heart of VPR lies the information retrieval pipeline, which pivots on extracting and comparing feature vectors from images to determine their similarity. This process bifurcates into two distinct methodologies: global and local descriptor extraction.

Global descriptors [2, 3, 4, 5] compress the entire image into a single feature vector, leveraging Euclidean distance in the vector space to gauge similarity. While this approach is lauded for its speed, enabling rapid processing times, it often needs to improve accuracy. Conversely, local descriptors [6, 7, 8] extract a set of features for one image, each representing localised information. The similarity evaluation among these descriptors commonly employs the Random sample consensus (RANSAC) algorithm, offering enhanced accuracy at the expense of processing speed. This dichotomy presents a significant challenge, particularly for time-sensitive applications wherein the balance between speed and accuracy is paramount.

The prevailing practice in VPR involves an initial generation of candidate locations using global descriptors, followed by meticulous reranking with local descriptors to refine accuracy. However, this reranking phase is still a time-consuming bottleneck, undermining the system's efficiency.

1.2 Motivation

The Feature matching using RANSAC is the most time-consuming part of local descriptor reranking. This matching process can be divided into two image sets: queries and references. After extracting local features from all the images, cross-matching occurs between the items in the query and reference sets. Thus, the cross-matching has a high time complexity:

$$O(D(F) \times N(F) \times N(Q) \times N(R))$$

where $D(F)$ is the feature dimension, $N(F)$ is the feature amount, $N(Q)$ is the number of queries, and $N(R)$ if the number of references.

Zhang et al. [9] find that with an anchor-patch matching structure called attentive patch, the speed could be significantly increased by much fewer descriptors to match for RANSAC with a little loss of accuracy. The performance of this architecture highly depends on the quality of the anchors. Although the attentive patch performed well in most datasets, it struggled with some datasets like SPED. Thus, this project aimed to identify its reason and propose solutions.

1.3 Objectives

This project aims to identify the limitations of the attentive patch algorithm used in local descriptor matching and explore solutions to address these challenges. Specifically, by examining and enhancing the algorithm's performance, the project seeks to improve the overall efficiency of local descriptor reranking. This work applies both the anchor selection strategy and adaptive thresholding as part of its approach to enhance the attentive patch algorithm.

1.4 Scope

This project's scope is centred on identifying the limitations of the current algorithm used in local descriptor matching and providing insights and understanding into this process. While the project includes evaluating anchor selection strategies and adaptive threshold setting, the primary focus is on enhancing the understanding of local descriptor matching and its implications for algorithm efficiency and improving it. The impact of any proposed improvements is assessed in terms of speed and accuracy across various environments, compared to traditional local descriptor reranking approaches regarding processing speed, accuracy, and computational efficiency.

1.5 Contributions

- Problem Identification: We identified two problems with the attentive patch algorithm and provided an analysis of their sources.
- Anchor Selection Strategy Experimentation: We experimented with various anchor selection strategies and analysed the pros and cons of each.
- Adaptive Threshold Method: We proposed an adaptive threshold method to enhance the algorithm's performance.
- Algorithm Improvement: Our proposed methods helped the local descriptor VPR architecture outperform the original design in speed and accuracy.

1.6 Report Orgnisation

Our report comprehensively examines local descriptor matching within Visual Place Recognition (VPR), organized into a sequence of sections designed for clarity and depth. Initially, the report introduces the context and motivations for this study, establishing the significance of local descriptor matching in VPR and stating the research objectives. The section on related work provides a foundation, discussing Simultaneous Localization and Mapping (SLAM), VPR, established feature extraction methods, RANSAC-based cross-matching, and the Attentive Patch algorithm, elucidating their relevance and existing contributions.

We then identify and analyze the current challenges in local descriptor matching, presenting a theoretical backdrop against which the limitations of current methods are examined. This leads us to a detailed explanation of our methodology, where we propose a novel anchor selection strategy and an adaptive threshold mechanism aimed at refining the matching process.

The implementation section follows, offering insight into the system setup, including the reasoning behind our hyperparameter choices, ensuring the reader understands the practical execution of our proposed solutions. Our experimental results are then thoroughly discussed, showcasing the empirical evidence of performance improvements gained through our approach.

The report concludes with a critical discussion that evaluates the strengths and weaknesses of the methods used. This section reflects on the problems identified and the potential for future research, suggesting pathways for further advancements in the field. Each section is interconnected, ensuring the report progresses logically from foundational concepts to innovative approaches and prospective developments.

Chapter 2

Related Works

2.1 Simultaneous Localization and Mapping (SLAM)

Simultaneous Localization and Mapping (SLAM) [1] is a cornerstone technology in robotics, pivotal for enabling autonomous navigation and comprehensive environmental understanding. Research in this domain spans several decades, branching into two principal categories: visual SLAM [10, 11] and LiDAR SLAM [12]. While LiDAR-based approaches are celebrated for their precision, their associated costs have steered a broader adoption towards Visual SLAM, given its practicality and versatility across diverse applications.

While the robot or any intelligent agent is moving, the sensor will provide data from the real world in a live stream for algorithms to perform further processes. In visual SLAM, the data is usually mono RGB, stereo RGB, or RGB-D video stream depending on the type of sensor. After the keypoint extraction and triangulation calculation, the relative pose transformation between adjacent steps and move trajectory can be estimated and a 3D scene representation (e.g., point cloud [10], mesh [13], neural implicit representation [14, 15], 3D Gaussian Splatting [16, 17]) could be generated. This is called visual odometry. However, this method, though efficient, is prone to accumulating drift errors over time, which can significantly skew the estimated trajectory and the constructed map.

To counteract drift errors, loop closure detection [18] has emerged as a crucial advancement within the SLAM research community. This technique involves recognising when the camera revisits a previously mapped area, thereby enabling the correction of drift errors by introducing new constraints within the pose graph. This graph, which chronicles the camera's trajectory, is subject to continuous optimisation, enhancing pose estimation and mapping accuracy.

2.2 Visual Place Recognition

Visual Place Recognition (VPR) is a critical component of robotics's loop closure detection process, acting as a linchpin for effective navigation and mapping. At the core of VPR lie descriptors, which are instrumental in representing places with high fidelity. Descriptors are bifurcated into two principal categories: global descriptors [2, 3, 4, 5] and local descriptors [19, 7, 20], each with its distinct role in the VPR process.

Global descriptors have been pivotal in encapsulating an entire scene's characteristics into a singular, high-dimensional feature vector. With the integration of neural networks, the focus has shifted towards leveraging feature maps for local information encapsulation, culminating in trainable aggregation layers like NetVLAD [2] and Generalized Mean Pooling (GeM) [21].

Local descriptors are designed to capture geometric and contextual details within a localised area, offering a fine-grained representation of the environment. The efficacy of these descriptors is closely tied to the process of keypoint selection, which identifies the most descriptive and consistent points within images. This selection is critical, fundamentally influencing the precision and reliability of the resulting descriptors. Two prevalent methodologies, detect-then-describe [19, 22] and describe-then-detect [23] outline the approaches to keypoint detection and descriptor extraction. Traditional methods like Scale-Invariant Feature Transform (SIFT) [19] embody the detect-then-describe paradigm, prioritising the identification of key points before descriptor aggregation. Conversely, neural network advancements have prompted a shift towards describe then detect strategies, exemplified by D2-Net [24] and its use of triplet loss for feature robustness and soft detection via Non-Maximum Suppression (NMS).

2.3 Feature Extraction

SuperPoint [6] leverages deep learning [25] techniques to revolutionize the extraction of local descriptors and keypoints. This neural network employs an end-to-end approach, where both feature detection and description are jointly optimized during the training process. Driven by data, the network updates its parameters through back-propagation [26], enhancing its ability to discern and describe salient features within an image.

One of the distinctive advantages of SuperPoint is its origin in learning from synthetic data. This foundational training allows the network to develop an acute sensitivity to geometric patterns and structures, distinguishing it from other feature detectors that may rely more on textural cues. Such geometry-centric feature detection ensures that the keypoints are not just randomly scattered across the image, but rather concentrated around the most structurally informative parts of the scene.

The capacity for SuperPoint to learn geometrically meaningful features directly aligns with the objectives of our project, where robust and repeatable local descriptor extraction is paramount. The network's adeptness in highlighting points of interest based on their geometric significance rather than mere intensity variations provides a substantial advantage for applications in place recognition, where consistency across different viewpoints and conditions is crucial. This intrinsic attribute of SuperPoint, rooted in its learning methodology and deep learning framework, is the principal rationale behind its selection as the cornerstone of our local descriptor extraction process.

2.4 RANSAC based cross-matching

In feature-based image matching, the initial step involves establishing correspondences between local descriptors from a query image and a reference image. The goal is to identify the most similar descriptor in the reference image for each descriptor in the query image, quantified by cosine similarity. For descriptor f_{q_i} from the query image and descriptor f_{r_j} from the reference image, the similarity is calculated as:

$$\text{similarity}(f_{q_i}, f_{r_j}) = \frac{f_{q_i} \cdot f_{r_j}}{||f_{q_i}||||f_{r_j}||} \tag{2.1}$$

A valid match requires that each descriptor from the query image is paired with a descriptor from the reference image such that they are each other's best match.

After establishing initial matches based on descriptor similarity scores, a significant challenge remains the presence of outliers among these matches. These outliers are often due to various discrepancies, such as noise or changes in perspective, which can introduce substantial errors in estimating the homography matrix.

RANSAC addresses this by iteratively selecting random subsets of matched point pairs and estimating the homography matrix that best aligns the points from the reference image to the query image. For each trial, RANSAC applies the estimated tomography matrix H to all matched points and counts the number of inliers—pairs that align within a certain tolerance level, indicating a low error of the transformation.

The algorithm repeats this process multiple times with a different random subset of matches to find the homography matrix that yields the maximum number of inliers. This consensus approach effectively filters out mismatched pairs, as only those consistently fit the estimated transformation across trials are retained.

The resulting homography matrix H relates corresponding points r_i from the reference image and q_i from the query image under the projective transformation that is expected to hold for different views of the same scene.

$$s \begin{bmatrix} q'_{ix} \\ q'_{iy} \\ 1 \end{bmatrix} = H \begin{bmatrix} r_{ix} \\ r_{iy} \\ 1 \end{bmatrix} \tag{2.2}$$

where (q'_{ix}, q'_{iy}) are the coordinates of the projected point in the query image corresponding to the point (r_{ix}, r_{iy}) in the reference image, and s is a scaling factor. This projective transformation is central to the process of matching features between two images of the same scene.

The RANSAC algorithm aims to find the homography matrix that results in the maximum number of inliers, thereby ensuring the most accurate correspondence of features between the two images.

2.5 Attentive Patch

The Attentive Patch algorithm selects distinct anchor features from the query image for efficient matching. It identifies the most similar feature in the reference image for each anchor, focusing on likely matches to reduce errors. An $n \times n$ patch around each anchor pair is examined, retaining only the most relevant features based on a threshold. These refined features are then used for RANSAC matching to exclude outliers and enhance accuracy. The similarity score, determined by the ratio of matched pairs after RANSAC to the total query features, quantifies the matching success, providing a streamlined approach for improved place recognition.

A more strict mathematical definition for the above progress could be:

Let $Q = \{q_1, q_2, \ldots, q_m\}$ and $R = \{r_1, r_2, \ldots, r_n\}$ be the sets of features for the query and reference images, respectively. Define a function $f : Q \to A$ that selects a subset of anchors $A \subseteq Q$ from the query feature set using an anchor selection strategy.

For each anchor $a_i \in A$, the best-matched feature $r_j \in R$ is determined by:

$$r_j = \underset{r \in R}{\operatorname{argmax}} \operatorname{similarity}(a_i, r) \tag{2.3}$$

If the similarity of the matched pair (a_i, r_j) is greater than a threshold δ_1, the anchor is kept; otherwise, it is discarded:

$$\text{Keep } a_i \text{ if } \operatorname{similarity}(a_i, r_j) > \delta_1 \tag{2.4}$$

For each remaining anchor a_i and its matched feature r_j, select a patch of size $n \times n$ centred around a_i in the query image and r_j in the reference image. Let P_{a_i} and P_{r_j} be the sets of features in these patches.

The pairs (a_i, r_j) are kept if their patch similarity is greater than another threshold δ_2:

$$\text{Keep } (a_i, r_j) \text{ if cosine\_similarity}(P_{a_i}, P_{r_j}) > \delta_2 \tag{2.5}$$

Use the remaining matched pairs (a_i, r_j) as input to the RANSAC algorithm to calculate the homography matrix. After RANSAC, only a few matched pairs will remain.

The similarity score is calculated as the number of matched pairs remaining after RANSAC divided by the total number of input query features m:

$$\text{Similarity Score} = \frac{\text{Number of matched pairs after RANSAC}}{m} \tag{2.6}$$

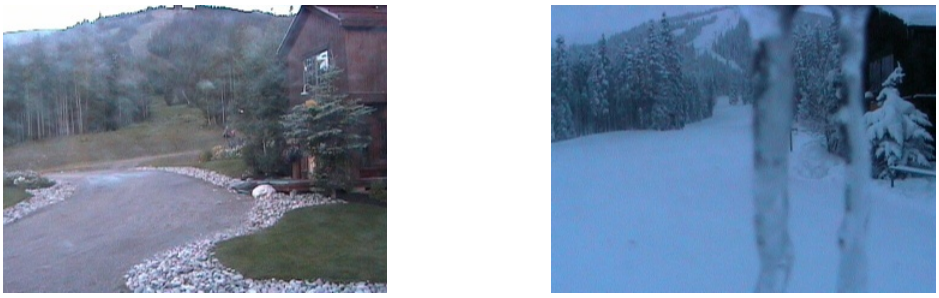
This description provides a mathematical representation of the attentive patch-matching process, emphasising the use of equations for feature matching and patch similarity.

Chapter 3

Methods

3.1 Problem Identification

The Attentive Patch algorithm encounters significant hurdles when tested against the Season, Weather, and Illumination changing (e.g. SPED) dataset. Thus, the SPED dataset presents a considerable challenge for achieving consistent place recognition.



(a) Query

(b) Reference

Figure 3.1: An example of the data in the SPED dataset. The query (on the left) and reference (on the right) images represent the same place in different weather.

For instance, consider the scenario in 3.2 where the query image is captured in spring, while the reference image is taken in winter. In this case, compared with the correct prediction made by cross-matching in figure 3.3, the attentive patch algorithm tends to overlook crucial landmarks, such as a tree in the centre of the scene, which remains a consistent point of reference despite seasonal changes. Instead, the algorithm disproportionately focuses on less relevant areas like the sky, which may appear markedly different between the two images due to weather variations.

This skewed attention results in a high matching score for the query image, leading to an incorrect prediction. However, the matching score between the reference and query images is lower, indicating a misalignment in the algorithm's focus. This example underscores the need to improve the attentive patch algorithm to accommodate environmental changes better and ensure more reliable scene recognition.

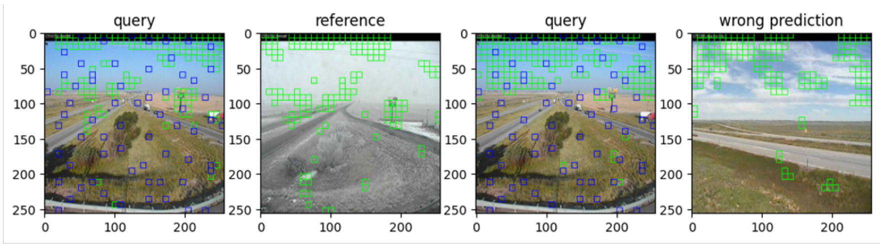


Figure 3.2: Wrong prediction for attentive patch. The blue squares are anchors and the green squares are matched patches using attentive-patch. The first and third images are the same query images but with different attention visualizations. The second image is the reference image and the fourth is the wrong prediction. In this case, the similarity between the query and wrong prediction is higher than the query and reference.

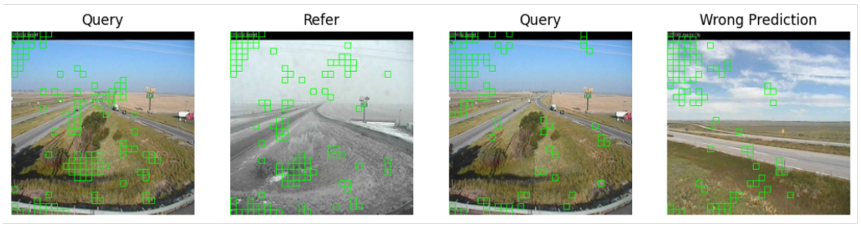


Figure 3.3: Cooresponding result for cross-matching. The similarity for query and refer is higher than query and wrong prediction here. The labels and images are just for comparison with the attentive patch

Through the preliminary analysis of the attentive patch algorithm, we have identified two significant issues that impact its performance in local descriptor matching:

- The algorithm struggles to detect critical, distinct elements within the scenes. This lack of detection of discriminative features significantly undermines the algorithm's capacity to accurately recognise locations amidst environmental changes.

- There is an apparent inclination within the algorithm to focus excessively on repetitive features. Despite their prevalence, architectural patterns or road markings do not offer unique location identifiers. The algorithm's disproportionate attention to these repetitive features leads to an inefficient emphasis on less informative aspects of the scene.

These problems were identified by carefully examining the algorithm's behaviour in various scenarios. While a detailed analysis and validation of these issues are presented in the experimental results section, it is important to note that these preliminary findings highlight areas for improvement in the algorithm's design and implementation.

3.2 Theoretically Analysis

To enhance the processing speed, two threshold-based matching are used in the attentive patch: equation 2.4 and equation 2.5. This one-way matching approach deems reference features as "matched" to those in the query if their similarity exceeds a predetermined threshold.

In the Attentive Patch algorithm, two specific cases arise due to threshold-based matching. The first case involves features q and r from the query and reference sets, respectively, where q exhibits the largest cosine similarity with r among all query descriptors, and r shows the largest similarity with q among all reference descriptors. Despite this mutual preference, the absolute similarity between r and q may fall below the threshold, rendering the pair unmatched:

$$\begin{aligned}
&\max_{q' \in Q} \text{similarity}(q', r) = \text{similarity}(q, r) \quad \text{and} \\
&\max_{r' \in R} \text{similarity}(q, r') = \text{similarity}(q, r) \quad \text{but} \\
&\text{similarity}(q, r) < \delta_1
\end{aligned}$$

The second case addresses the scenario of two highly similar patches, P_{q_i} and P_{r_j}, where each descriptor in P_{q_i} is very close to each descriptor in P_{r_j}. Although these patches exhibit high absolute similarity, establishing a one-to-one matching between descriptors becomes challenging. For instance, a descriptor i in P_{q_i} may have the highest similarity with descriptor j in P_{r_j}, but j might be more similar to a different descriptor k in P_{q_i}:

$$\forall p \in P_{q_i}, \exists q, q' \in P_{r_j} \text{ such that similarity}(p, q) > \text{similarity}(p, q') \quad \text{but}$$
$$\exists p' \in P_{q_i} \text{ with similarity}(p', q) > \text{similarity}(p, q)$$

To prove that the second case is likely to happen when the descriptors in the patches are all very close, we need to consider the nature of cosine similarity and the effect of closely spaced descriptors on this metric.

Cosine similarity measures the cosine of the angle between two vectors, which in this context are the descriptors of the features. It is defined as:

$$\text{similarity}(p, q) = \frac{p \cdot q}{\|p\|\|q\|}$$

When descriptors are very close to each other, their vectors in the feature space are nearly parallel, resulting in cosine similarities close to 1. This high similarity can lead to ambiguity in matching, as multiple descriptors in one patch may have nearly equal similarity scores with a descriptor in the other patch.

Formally, let's assume we have two patches P_{q_i} and P_{r_j} with descriptors that are very close to each other:

$$\forall p, p' \in P_{q_i}, \forall q, q' \in P_{r_j}, \quad \text{similarity}(p, q) \approx \text{similarity}(p', q') \approx 1$$

In this scenario, for a given descriptor $p \in P_{q_i}$, there might be multiple descriptors $q, q' \in P_{r_j}$ such that:

$$\text{similarity}(p, q) \approx \text{similarity}(p, q') \approx 1$$

Similarly, for a given descriptor $q \in P_{r_j}$, there might be multiple descriptors $p, p' \in P_{q_i}$ such that:

$$\text{similarity}(p, q) \approx \text{similarity}(p', q) \approx 1$$

This situation leads to the second case, where a one-to-one matching between descriptors becomes challenging due to the high similarity scores shared by multiple descriptor pairs. As a result, the algorithm may struggle to determine the best match for each descriptor, leading to potential inaccuracies in the matching process.

These cases illustrate the complexities and challenges in achieving precise and efficient feature matching in the Attentive Patch algorithm, highlighting the need for careful threshold selection and algorithmic refinement. The second case, in particular, is likely to occur when the descriptors in the patches are very close, leading to high similarity scores and potential ambiguities in matching.

In the cross-matching method, features from two images are matched based on their relative similarity without a strict threshold for absolute similarity values.

This flexibility allows the method to adapt to variations caused by seasonal, weather, and illumination changes, which might affect the absolute values of features representing the same object. As such, cross-matching is less prone to missing discriminative features that have lower absolute values due to environmental changes. This method's adaptability makes it particularly effective in dealing with the SPED dataset's challenges, where environmental variations are pronounced.

These findings suggest two primary reasons why the Attentive Patch algorithm may underperform in comparison to traditional cross-matching techniques, particularly on the SPED dataset:

- Missed Detection of Environmentally Variant Features: The algorithm's threshold-based matching is susceptible to missing discriminative features that undergo variations in appearance due to environmental changes, thereby reducing its ability to accurately recognise places across different conditions.

- Inefficiency in Handling Repetitive Features: The algorithm's propensity to prioritise features with high absolute similarity, irrespective of their uniqueness, leads to an inefficient allocation of attention. This approach fails to discriminate between truly informative matches and those involving repetitive, non-distinctive elements, impacting the overall accuracy of place recognition.

In light of these insights, further development and refinement of the Attentive Patch algorithm should focus on enhancing its matching mechanism to accommodate environmental variations better and discriminate more effectively against non-informative, repetitive features.

3.3 Anchor Selection Strategy

Optimising the anchor selection function f is a logical approach to address over-attention in non-important places. The original paper on the Attentive Patch algorithm proposed the self-similarity score-based anchor selection strategy for selecting discriminative anchors. In this work, we have designed and explored additional strategies to enhance the effectiveness of anchor selection. These strategies aim to ensure that the selected anchors are not only distinctive but also representative of critical features in the scene, thereby improving the overall accuracy and efficiency of the place recognition process.

3.3.1 Self-similarity Score-based Selection

Given a matrix of descriptors D representing the features of an image, the matrix is divided into grids of size $\frac{n}{8} \times \frac{n}{8}$, where $n \times n$ is the dimension of the original descriptor matrix. Within each grid G, the self-similarity score $S(d)$ of a descriptor d is computed by summing the similarity of d with all other descriptors d' in the same grid, $S(d) = \sum_{d' \in G, d' \neq d} \text{similarity}(d, d')$. The descriptor with the lowest self-similarity score, indicating its distinctiveness, is selected as the anchor for that grid, $a = \underset{d \in G}{\operatorname{argmin}} S(d)$. This process is repeated across all grids to identify a set of anchors A that are distinctive and distributed across the matrix, aiming to enhance the accuracy of place recognition by focusing on highly descriptive features.

3.3.2 High-Pass Filter-Based Selection

The convolutional high-pass filter method for anchor selection is based on the principle that areas of rapid change in an image, such as edges and textures, are often the most informative for place recognition.

The filter is applied to the image using convolution, a mathematical operation that combines two functions to produce a third function. In the context of image processing, convolution involves sliding the filter kernel (a small matrix) over the image and computing the sum of the element-wise products at each position. The result is a filtered image that highlights regions with significant variations.

Let I be the original image and F be the high-pass filter kernel. The filtered image I' is obtained by convolving I with F:

$$I'(x, y) = \sum_{u=-k}^{k} \sum_{v=-k}^{k} I(x - u, y - v) \cdot F(u, v)$$

where k is the radius of the filter kernel F, and $I'(x, y)$ is the value of the filtered image at position (x, y).

Anchors are selected from areas in I' where the filter response is non-zero. This strategy focuses on leveraging the inherent information within the texture and edges of the scene, hypothesising that these features contribute more significantly to effective place recognition.

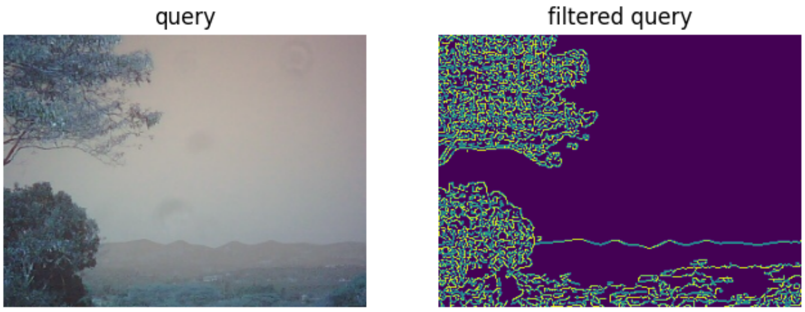


Figure 3.4: Left: A query image depicting a natural scene. Right: The same image after high-pass filtering to identify anchor candidates, with areas likely to contain distinctive features outlined.

3.3.3 Semantic Segmentation-Based Anchor Selection

The semantic segmentation-based method for anchor selection aims to mitigate the impact of repetitive features, such as large expanses of sky or water, on the algorithm's performance. By employing a segmentation model like SegFormer, the algorithm can distinguish between different semantic parts of the image. Mathematically, let I be the original image and $S(I)$ be the segmentation mask produced by the model, where each pixel in $S(I)$ corresponds to a semantic class label. The set of features $Q = \{q_1, q_2, \ldots, q_m\}$ extracted from the image is then filtered to exclude features that fall within regions labelled as sky or water:

$$Q' = \{q_i \in Q \mid S(I)(q_i) \notin \{\text{sky}, \text{water}, \ldots\}\}$$

The remaining features Q' are considered for anchor selection, focusing the algorithm on more discriminative image parts, such as buildings and mountains, which offer a higher likelihood of unique place identification. This approach leverages prior knowledge to guide the algorithm effectively, enhancing its ability to recognise distinct locations.

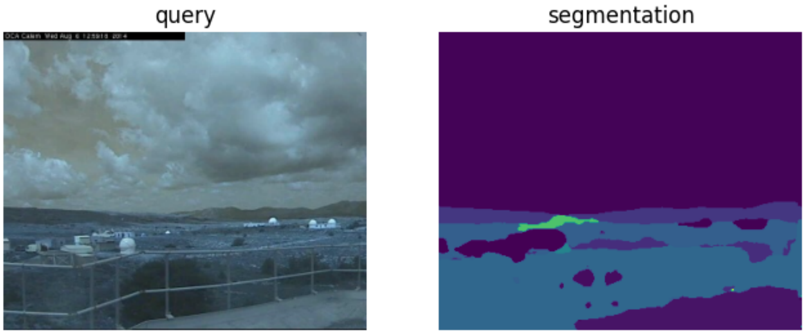


Figure 3.5: Left: Original query image used for Visual Place Recognition (VPR). Right: Segmentation mask applied to the image, filtering out non-essential features like sky and clouds to focus on more distinctive landmarks for VPR.

3.4 Adaptive threshold

The adaptive thresholding mechanism in the Attentive Patch algorithm dynamically adjusts the matching threshold $\delta(q_i)$ for each feature q_i based on its proximity to significant keypoints identified by the SuperPoint algorithm. This mechanism is mathematically defined as:

$$\delta(q_i) = \delta_{\text{high}} - (\delta_{\text{high}} - \delta_{\text{low}}) \cdot \min\left(1, \frac{\min_{k \in K} d(q_i, k)}{\tau}\right)$$

where $d(q_i, k)$ is the distance between feature q_i and keypoint k, τ is a predefined distance threshold, δ_{low} is the threshold value at keypoints, and δ_{high} is the largest threshold value reached when the distance to the nearest keypoint is at least τ. This formulation ensures that the threshold $\delta(q_i)$ gradually increases with the distance to the nearest keypoint, starting from δ_{low} at the keypoint and reaching δ_{high} at a distance of τ or greater.

This design addresses two key problems:

- Enhanced Detection of Critical Features: Near keypoints, where $\min_{k \in K} d(q_i, k) \leq \tau$, the threshold $\delta(q_i)$ is closer to δ_{low}, increasing sensitivity to potential matches. This ensures that critical features are more likely to be detected and matched.

- Reduction of Over-Detection in Repetitive Areas: In regions far from keypoints, where $\min_{k \in K} d(q_i, k) > \tau$, the threshold $\delta(q_i)$ reaches δ_{high}, reducing the likelihood of matching non-discriminative, repetitive features. This minimises over-detection and focuses the algorithm on more informative features.

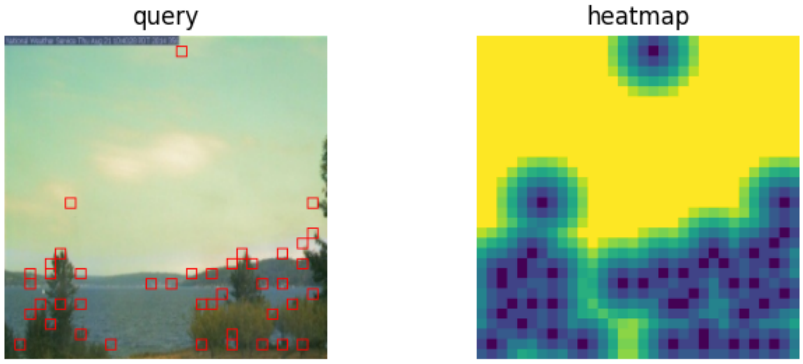


Figure 3.6: Left: Landscape with keypoints marked by red squares. Right: Heatmap indicating adaptive thresholds; areas with higher brightness denote increased thresholds.

3.5 Matching Score

The matching score in the Attentive Patch algorithm is defined using the cosine similarity between descriptors, which quantifies the similarity between two feature vectors. Mathematically, the matching score M between two descriptors q_i and r_j from the query and reference images, respectively, is given by:

$$M(q_i, r_j) = \frac{q_i \cdot r_j}{||q_i|| ||r_j||}$$

where $q_i \cdot r_j$ denotes the dot product of the vectors, and $||q_i||$ and $||r_j||$ are the Euclidean norms of the vectors. The cosine similarity ranges from -1 to 1, where 1 indicates that the vectors are identical, 0 indicates orthogonality, and -1 indicates opposite directions. In the context of feature matching, a higher cosine similarity score implies a higher degree of similarity between the features, making it a suitable metric for the matching score in the Attentive Patch algorithm.

Chapter 4

Implementation

4.1 Feature Extraction

The SuperPoint network is deployed to identify and describe salient features within the image. The initial step involves preprocessing the input image and resizing it to a 256x256 pixel resolution to ensure compatibility with the network's input specifications. SuperPoint is fine-tuned using a trio of hyperparameters crucial for its operation. The non-maximum suppression distance, denoted as *nms_dist*, is configured to a value of 4, allowing the network to discern and select keypoints by enforcing a minimum spatial separation. The *nn_thresh*, set at 0.7, dictates the acceptable L2 distance between descriptor pairs for a match to be considered valid, effectively acting as a gauge for match quality. Lastly, the *conf_thresh* is fixed at 0.015, serving as a confidence filter to retain only those keypoints deemed reliable by the network. With these parameters in place, the SuperPoint network processes the input to yield a structured output of 32x32 descriptors, each being a 256-dimensional vector that captures the essence of the image's local features. These descriptors form the foundation for the intricate process of keypoint selection and matching that follows.

4.2 Descriptor Matching

A threshold of 0.7 is established for anchor matching to ensure that only descriptors with a high degree of similarity are considered matches. A lower threshold of 0.55 is employed for patch matching, accounting for the slight variations expected in the local neighbourhood of descriptors. The patches themselves are defined as 7x7 squares centred on the anchors, providing a contextual area for evaluating the consistency of local features.

Homography, the transformation crucial for establishing the spatial relationship between matched features, is computed using OpenCV's *findHomography* function. The *ransacReprojThreshold* is set to 3, implying that a homographic model consensus requires at least three agreeing points to qualify as a match, ensuring robustness against outliers.

The system uses PyTorch to leverage computational efficiency, which inherently supports GPU acceleration. This choice allows for rapid matrix computations essential for descriptor matching in high-dimensional spaces. Performance and speed are further optimised by executing experiments on a hardware setup that includes a GPU 1080ti and a CPU Intel i7-7820x. This hardware configuration ensures the matching process is accurate and time-efficient, making it suitable for applications where rapid visual data processing is paramount.

4.3 Adaptive Threshold

In the adaptive threshold preprocessing code, a heatmap is constructed by first initialising a zero-valued matrix with dimensions corresponding to the predetermined grid size. This matrix is designed to record the threshold levels for feature matching. Anchors, which are keypoints of interest previously identified, are iterated over the entire grid. The minimum Euclidean distance to any of the anchors is calculated for each grid cell. Suppose this minimum distance is within a predefined range. In that case, the threshold value for that cell is interpolated between a minimum and a maximum threshold value, inversely proportional to the distance—closer cells have a lower threshold, making the matching process more sensitive to nearby features. We set the minimum value as 0.4 and the highest value as 0.9. For cells beyond this range, the threshold is set to a higher constant value, reducing sensitivity and the likelihood of matching repetitive, non-distinctive features. This process generates a heatmap where the gradient of threshold values reflects the spatial distribution of salient features, to optimise the matching stage for accuracy and computational efficiency. The completed heatmap is then stored, contributing to a collection of heatmaps that subsequently influence the descriptor-matching process.

Chapter 5

Results and analysis

5.1 Problem Identification

	common failed cases	unique failed cases	total number
attentive patch	57	69	126
cross-matching	57	7	64

Table 5.1: Comparison of Failed Cases Between Attentive Patch and Cross-Matching Methods. The table presents the distribution of common and unique failed cases, highlighting that while both methods share a similar count of common failures, Attentive Patch exhibits a significantly higher number of unique failed cases. This observation suggests that further investigation into the unique failures of Attentive Patch could provide valuable insights for algorithmic improvement.

In the initial experiment, we aimed to identify and compare the limitations of the Attentive Patch algorithm against the more traditional cross-matching method in the context of Visual Place Recognition (VPR). To achieve this, we meticulously recorded instances where each algorithm failed to match features in a controlled test environment correctly.

Using the established criteria for a matching failure, we ran both algorithms through the SPED dataset and categorised the failures into two groups: *common failed cases*, which are failures shared by both algorithms and *unique failed cases*, which are failures exclusive to a particular algorithm.

The outcome of this comparative analysis was then tabulated to provide a clear visualisation of the failure distribution between the Attentive Patch and cross-matching techniques. The results revealed that while the number of common failed cases was identical for both algorithms, the Attentive Patch had a substantially higher number of unique failed cases.

This discrepancy in unique failures between the two methods underscores a potential specificity in the shortcomings of the Attentive Patch algorithm. Investigating these unique failed cases further could shed light on inherent weaknesses within the Attentive Patch approach and thus inform future algorithm design improvements. This exploration stands as the first experiment in our ongoing research to refine and advance the capabilities of VPR systems.

5.2 Anchor Selection Comparison

Method	**Nordland**	**Pittsburgh**	**Cross Seasons**	**SPED**	**ESSEX3IN1**
Self Score	0.281 (775/2760)	0.982 (982/1000)	0.989 (189/191)	0.792 (481/607)	0.948 (199/210)
High-pass Filter	0.354 (977/2760)	0.976 (976/1000)	0.994 (190/191)	0.828 (503/607)	0.952 (200/210)
Semantic Segmentation	0.313 (863/2760)	0.979 (979/1000)	0.984 (188/191)	0.794 (482/607)	0.952 (200/210)
Cross Matching	0.537 (1482/2760)	-	1.0 (191/191)	0.866 (526/607)	0.957 (201/210)

Table 5.2: Performance comparison across various methods and datasets, detailing recall rates above and successful-to-total match counts below.

5.2.1 Self-similarity Score-based Selection

The strategy outlined in the original Attentive Patch paper is utilised in the self-similarity score-based anchor selection experiment as the baseline.

The experimental setup was configured to reproduce the conditions under which the original results were obtained. We analysed a comprehensive set of images, employing the same preprocessing and feature extraction steps as the original study to ensure that any deviation in performance could be attributed to the algorithm rather than external variables.

Throughout this process, we meticulously logged instances where the anchor selection failed to recognize or prioritise critical features within an image. We also noted cases where the strategy erroneously gave weight to repetitive and non-discriminative elements, such as vast areas of sky or water. These shortcomings provided a clear insight into the limitations of the self-similarity score-based method.

By conducting this experiment, we aimed to establish a solid benchmark against which we could measure the performance of our proposed enhancements. The insights gleaned from the shortcomings of the self-similarity score-based strategy have been instrumental in guiding our improvements and have served as a foundational comparison point throughout our research.

5.2.2 High-Pass Filter-Based Selection

During the exploration of high-pass filter-based anchor selection, the experiment sought to evaluate its efficacy compared to the baseline. The high-pass filter method targets more distinct changes in the image, theorising that these areas hold more informative features for accurate place recognition.

We applied this strategy across five diverse datasets, carefully designed to challenge the algorithm with various environmental and structural complexities. The performance was quantitatively assessed, with the high-pass filter approach outperforming the baseline self-similarity score-based strategy in four out of the five datasets as shown in table 5.2.

Visual inspection of the results revealed that the high-pass filter method significantly reduced the number of features selected in typically uninformative areas, such as homogenous sky or water surfaces. This tailored focus on more variable regions of the image directly addressed one of the primary limitations noted in the baseline method.

Notably, this approach corrected 55 out of 126 incorrect predictions attributed to the self-similarity score-based selection, underscoring the practical improvements offered by employing a high-pass filter for anchor selection. The experiment's findings emphasise the potential of this method in enhancing the overall accuracy of Visual Place Recognition systems by better discriminating between informative and non-informative areas within an image.

5.2.3 Semantic Segmentation-Based Anchor Selection

In the set of experiments centred on semantic segmentation-based anchor selection, the goal was to investigate the impact of removing homogeneous regions on the performance of the anchor selection process. By incorporating semantic segmentation, the strategy aimed to exclude uninformative parts such as sky or water, which are typically invariant and not useful for the purposes of Visual Place Recognition (VPR).

The results demonstrated that semantic segmentation effectively removed homogenous parts, often more rigorously than the high-pass filter approach. It solved 24% of the failed cases in the self-similarity score anchor selection strategy.

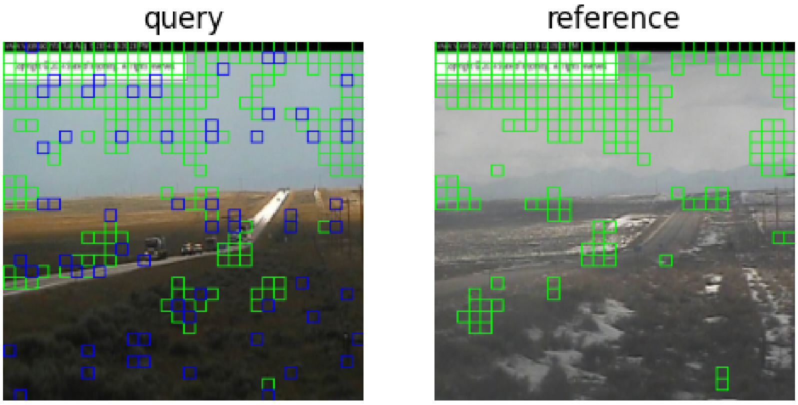


Figure 5.1: Query image with keypoints highlighted, demonstrating a misfocus on the sky. Right: The reference image shows a better distribution of keypoints for matching.

The images 5.1 depict a case study where the self-similarity score-based anchor selection process is prone to errors. In the query image on the left, the algorithm erroneously selects numerous keypoints in the sky area—indicated by the blue squares—which are less informative due to their homogeneous nature. Conversely, the reference image on the right, marked by green squares, displays a more advantageous spread of keypoints, including the land and infrastructure, providing more distinct features for accurate matching. This illustrates the tendency of the self-similarity score approach to overemphasise less useful areas, potentially leading to suboptimal matching performance.

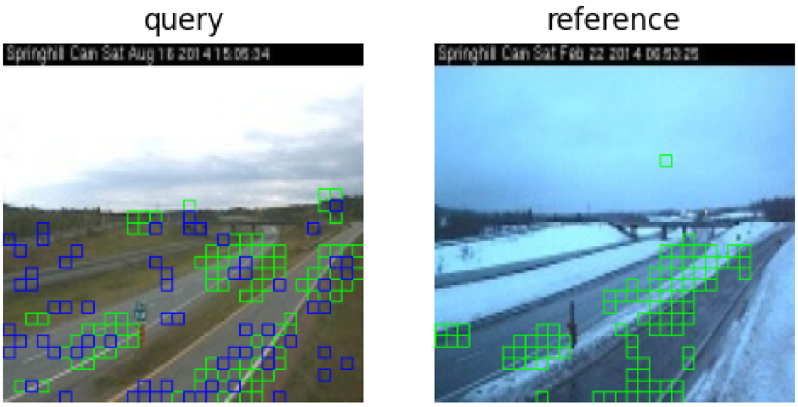

query
reference
Springhill Cam Sat Aug 16 2014 15:06:34
Springhill Cam Sat Feb 22 2014 06:53:25

Figure 5.2: Left: Query image with keypoints, showing a preference for ground features. Right: Reference image with a focus on structural landmarks for feature matching.

The images 5.2 showcase the effectiveness of semantic segmentation-based anchor selection. In the query image on the left, semantic segmentation has resulted in keypoints (blue squares) predominantly positioned over the ground and structural elements rather than the expansive sky. The reference image on the right highlights a similar pattern, with keypoints (green squares) focused on areas that offer more distinct, stable features for matching, like the road and its surroundings, despite snow. This demonstrates the method's capacity to concentrate on an image's most informative parts, enhancing the matching process's precision and reliability across seasonally diverse conditions.

However, the strategy did not uniformly yield positive outcomes. The segmentation algorithm's limitations became apparent in certain instances, particularly with larger and more diverse place datasets. These limitations manifested in two notable problems.

Firstly, the segmentation algorithm demonstrated inadequate performance across expansive datasets, leading to potential misdirection in anchor selection. The misclassification of segments within these datasets could result in the omission of important features or the inclusion of irrelevant ones, diminishing the reliability of the subsequent matching process.

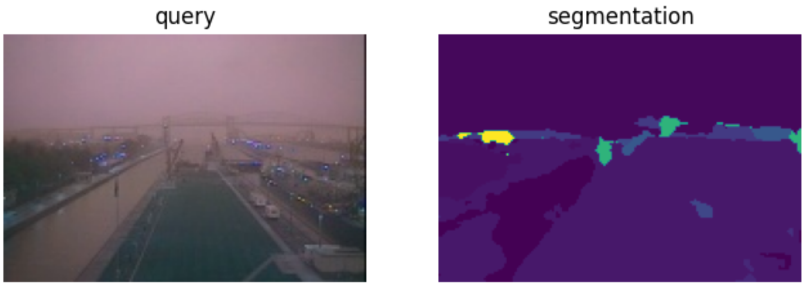


Figure 5.3: Left: Original query image with key landmarks. Right: Resulting segmentation highlighting inaccuracies in place data, potentially misguiding the anchor selection process.

Secondly, while removing vast areas of sky and water could streamline the matching process, it was observed that this also led to the loss of substantial global structural information within the images. Since such information can be vital for recognising a place, especially under varying conditions, its absence could degrade the system's performance.

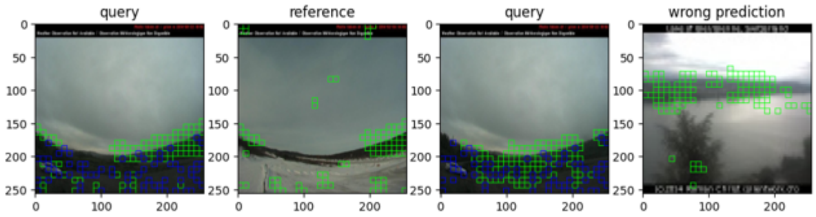


Figure 5.4: Comparison of keypoint distribution in query and reference images, with a subsequent wrong prediction due to segmentation-based anchor selection.

The images 5.4 demonstrate a scenario where segmentation-based anchor selection leads to incorrect matching. The first and third images show the query keypoints, while the second image displays reference keypoints, all marked in blue. The keypoints focus on specific segments of the images, likely identified as landmarks. However, the last image reveals a wrong prediction, with green squares indicating matched points. This error is attributed to the segmentation process, which has disproportionately focused on limited image regions, causing a mismatch during feature comparison and highlighting a potential drawback of relying solely on segmentation for anchor selection. The case illustrates the need for a more balanced approach that can adapt to the comprehensive structure of the scene.

These challenges highlighted by the semantic segmentation experiments provided a clear rationale for developing an adaptive thresholding approach. By dynamically adjusting the thresholds for local descriptor matching, we envisioned a method that would retain the benefits of segmenting out homogeneous regions without losing critical global contextual information. This adaptive method would ideally allow for the nuanced inclusion of regions like the sky and water when they contribute to the distinctiveness of a place, thereby enhancing the VPR system's robustness and accuracy.

5.2.4 Speed Analysis

Method	**Nordland**	**Pittsburgh**	**CrossSeasons**	**SPED**	**ESSEX3IN1**
Self Score	91.903 0.003 91.823	191.418 0.007 190.868	2.945 0.002 2.904	4.159 0.002 4.125	3.319 0.003 3.260
High-pass Filter	95.195 0.003 95.115	179.660 0.007 179.082	2.599 0.002 2.533	1.640 0.002 1.672	3.253 0.003 3.180
Semantic Segmentation	93.161 0.003 93.083	184.070 0.007 183.253	3.494 0.002 3.445	1.767 0.002 1.740	4.020 0.003 3.943
Cross Matching	870.031 0.003 869.950	- - -	5.625 0.002 5.574	19.220 0.002 19.181	8.287 0.003 8.228

Table 5.3: Speed analysis across different methods and datasets, presenting the average time per query, average encoder time, and average matching time per query, respectively.

The data reveals a compelling efficiency advantage of the anchor selection strategies over the traditional cross-matching technique. When examining the average time per query, the adaptive anchor selection methods consistently demonstrate lower time usage across the board. This indicates a more streamlined approach, reducing computational overhead significantly when compared to the cross-matching method.

Interestingly, on datasets such as SPED, which may present more complex visual environments, the anchor selection strategies not only surpass the cross-matching in terms of efficiency but also outperform the original algorithm. This efficiency is likely due to the focused nature of the anchor selection strategies, which, by design, prioritise specific features and reduce the number of comparisons needed during the matching process.

The marked speed improvement of these strategies suggests a robustness that could translate into practical benefits in real-world applications of VPR, especially in scenarios where rapid processing is essential. This data supports the hypothesis that intelligent selection of features, whether through attentive methods or even random selection complemented with adaptive thresholds, can yield significant time savings without compromising the accuracy and reliability of place recognition tasks.

The evaluation of time consumption across different stages of the algorithm reveals a significant insight: the RANSAC homography estimation step emerges as the most time-consuming element in both the cross-matching and Attentive Patch methods. However, there's a stark contrast in the duration spent on this process between the two approaches.

Cross-matching requires, on average, a substantially longer time of 16.70 seconds to complete the RANSAC step compared to just 1.31 seconds for the Attentive Patch. This discrepancy can be primarily attributed to the number of processed features: cross-matching funnels roughly 235 features into the RANSAC algorithm, whereas the Attentive Patch sends a more curated set of approximately 76 features.

The reduction in the number of features directly translates to faster processing times for the Attentive Patch, implying that RANSAC's efficiency is heavily dependent on the input feature count. Unlike matrix calculations for descriptor matching—where both methods are relatively close in timing, with cross-matching at 2.26 seconds and Attentive Patch at 2.19 seconds—RANSAC does not benefit from GPU acceleration. Hence, the performance optimisation for RANSAC lies in minimising the number of input features, which is a strength of the Attentive Patch method.

This data clearly underscores the need for strategies that can effectively limit the feature set without compromising the matching quality, as this is the main lever for reducing computation times without GPU acceleration capabilities for RANSAC.

5.3 Adaptive Threshold Experiment result

In the experiments assessing the efficacy of the adaptive thresholding mechanism, we specifically examined the impact of coupling random anchor selection with this novel approach. The findings indicated a significant increase in accuracy, improving from a baseline of 0.792 to 0.823. This enhancement in performance underlines the potential of adaptive thresholding to refine the feature-matching process in Visual Place Recognition systems effectively.

Remarkably, implementing the adaptive threshold advanced the results beyond those achieved by the semantic segmentation anchor strategy and delivered outcomes comparable to the high-pass filter method. This suggests that the adaptive thresholding mechanism, even when applied to randomly selected anchors, can discern and prioritise the most informative features for matching.

These experimental results advocate for the strength of adaptive thresholding as a robust strategy that can leverage the existing feature set more efficiently. By dynamically adjusting the threshold based on feature quality and distribution, it reduces reliance on more complex preprocessing methods without compromising and, in some cases, enhances the overall performance. This approach presents a promising avenue for further exploration and optimisation within the field of VPR.

Chapter 6

Limitation and Future Work

A key limitation of the current implementation lies in the efficiency of the adaptive threshold preprocessing computation. The computationally intensive process may lead to suboptimal performance, particularly in real-time applications or when processing large datasets. The reliance on rule-based techniques for generating heatmap also presents constraints, as they may not capture the complex patterns and variations in diverse environments.

To address these limitations, future work will explore two main avenues. First, efforts will be directed towards optimizing the computation of adaptive thresholds, possibly through algorithmic enhancements or leveraging parallel processing architectures. Acceleration could also be achieved by training a neural network to approximate this process, thereby reducing the computational overhead. Second, developing an end-to-end learning-based approach for heatmap generation is proposed. By utilizing deep learning techniques, the system can potentially learn more robust and intricate patterns, leading to improved accuracy and reliability in heatmap generation for Visual Place Recognition tasks.

Chapter 7

Conclusion

The project embarked on an ambitious objective to refine the process of matching local descriptors, a cornerstone in the functionality of Visual Place Recognition (VPR) systems. The endeavour was motivated by enhancing VPR in increasingly complex and varied environments. After a thorough examination, two significant shortcomings in the existing attentive patch algorithm were uncovered: a failure to consistently detect critical, distinct elements, and a propensity to over-focus on non-discriminative, repetitive features.

The project successfully designed and implemented an innovative anchor selection strategy and an adaptive thresholding mechanism to address these challenges. The newly established methods have been empirically validated, showcasing a notable outperformance over the traditional algorithm. This accomplishment underscores the potential of tailored solutions to improve the accuracy and reliability of VPR systems.

Despite these advances, the project was not without its constraints. The pre-processing phase of the adaptive thresholding proved to be laborious and computationally demanding, raising concerns about its viability in time-sensitive applications. Moreover, while methodologically sound, the rule-based approach to heatmap generation was based on strong presuppositions and did not fully exploit the rich tapestry of information that visual data offers.

In light of these limitations, future work is proposed to leverage learning-based algorithms. Such methodologies promise to address both the efficiency of adaptive thresholding and the depth of data utilization. The potential benefits of incorporating machine learning are manifold: they range from heightened processing speeds, thanks to models that can learn to approximate complex computations, to a more profound assimilation of scene features, resulting from the model's ability to learn from a wide spectrum of data.

By adopting a learning-based approach, future iterations of the VPR system can potentially self-adjust to a variety of environmental cues, making strides towards an autonomous system that is both agile and accurate. This evolution will not only pave the way for more sophisticated VPR systems but also contribute to the broader field of computer vision, where the accurate recognition of places forms a foundational pillar.

Bibliography

[1] Cesar Cadena et al. “Simultaneous Localization And Mapping: Present, Future, and the Robust-Perception Age”. In: *CoRR* abs/1606.05830 (2016). arXiv: 1606.05830. URL: http://arxiv.org/abs/1606.05830.

[2] Relja Arandjelović et al. *NetVLAD: CNN architecture for weakly supervised place recognition*. 2016. arXiv: 1511.07247 [cs.CV].

[3] Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. *MixVPR: Feature Mixing for Visual Place Recognition*. 2023. arXiv: 2303.02190 [cs.CV].

[4] Gabriele Berton, Carlo Masone, and Barbara Caputo. *Rethinking Visual Geo-localization for Large-Scale Applications*. 2022. arXiv: 2204.02287 [cs.CV].

[5] Gabriele Berton et al. *EigenPlaces: Training Viewpoint Robust Models for Visual Place Recognition*. 2023. arXiv: 2308.10832 [cs.CV].

[6] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. *SuperPoint: Self-Supervised Interest Point Detection and Description*. 2018. arXiv: 1712.07629 [cs.CV].

[7] Peter Hviid Christiansen et al. *UnsuperPoint: End-to-end Unsupervised Interest Point Detector and Descriptor*. 2019. arXiv: 1907.04011 [cs.CV].

[8] Xinjiang Wang et al. *FeatureBooster: Boosting Feature Descriptors with a Lightweight Neural Network*. 2023. arXiv: 2211.15069 [cs.CV].

[9] Dongshuo Zhang, Meiqing Wu, and Siew Kei Lam. “Training-Free Attentive-Patch Selection for Visual Place Recognition”. In: Oct. 2023, pp. 9169–9174. DOI: 10.1109/IROS55552.2023.10342347.

[10] Raul Mur-Artal, J. M. M. Montiel, and Juan D. Tardos. “ORB-SLAM: A Versatile and Accurate Monocular SLAM System”. In: *IEEE Transactions on Robotics*

31.5 (Oct. 2015), pp. 1147–1163. ISSN: 1941-0468. DOI: 10.1109/tro.2015.2463671. URL: http://dx.doi.org/10.1109/TRO.2015.2463671.

[11] Tong Qin, Peiliang Li, and Shaojie Shen. "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator". In: *IEEE Transactions on Robotics* 34.4 (2018), pp. 1004–1020. DOI: 10.1109/TRO.2018.2853729.

[12] Ji Zhang and Sanjiv Singh. "LOAM : Lidar Odometry and Mapping in real-time". In: *Robotics: Science and Systems Conference (RSS)* (Jan. 2014), pp. 109–111.

[13] Angela Dai et al. *BundleFusion: Real-time Globally Consistent 3D Reconstruction using On-the-fly Surface Re-integration*. 2017. arXiv: 1604.01093 [cs.GR].

[14] Edgar Sucar et al. *iMAP: Implicit Mapping and Positioning in Real-Time*. 2021. arXiv: 2103.12352 [cs.CV].

[15] Zihan Zhu et al. *NICE-SLAM: Neural Implicit Scalable Encoding for SLAM*. 2022. arXiv: 2112.12130 [cs.CV].

[16] Nikhil Keetha et al. *SplaTAM: Splat, Track & Map 3D Gaussians for Dense RGB-D SLAM*. 2023. arXiv: 2312.02126 [cs.CV].

[17] Hidenobu Matsuki et al. *Gaussian Splatting SLAM*. 2023. arXiv: 2312.06741 [cs.CV].

[18] Konstantinos A. Tsintotas, Loukas Bampis, and Antonios Gasteratos. "The Revisiting Problem in Simultaneous Localization and Mapping: A Survey on Visual Loop Closure Detection". In: *IEEE Transactions on Intelligent Transportation Systems* 23.11 (Nov. 2022), pp. 19929–19953. ISSN: 1558-0016. DOI: 10.1109/tits.2022.3175656. URL: http://dx.doi.org/10.1109/TITS.2022.3175656.

[19] D.G. Lowe. "Object recognition from local scale-invariant features". In: *Proceedings of the Seventh IEEE International Conference on Computer Vision*. Vol. 2. 1999, 1150–1157 vol.2. DOI: 10.1109/ICCV.1999.790410.

[20] Xiaoming Zhao et al. *ALIKE: Accurate and Lightweight Keypoint Detection and Descriptor Extraction*. 2022. arXiv: 2112.02906 [cs.CV].

[21] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. "Particular object retrieval with integral max-pooling of CNN activations". In: *arXiv preprint arXiv:1511.05879* (2015).

[22] Jerome Revaud et al. *R2D2: Repeatable and Reliable Detector and Descriptor*. 2019. arXiv: 1906.06195 [cs.CV].

[23] Vassileios Balntas et al. "Learning local feature descriptors with triplets and shallow convolutional neural networks". In: Jan. 2016, pp. 119.1–119.11. DOI: 10.5244/C.30.119.

[24] Mihai Dusmanu et al. *D2-Net: A Trainable CNN for Joint Detection and Description of Local Features*. 2019. arXiv: 1905.03561 [cs.CV].

[25] Jürgen Schmidhuber. "Deep learning in neural networks: An overview". In: *Neural Networks* 61 (Jan. 2015), pp. 85–117. ISSN: 0893-6080. DOI: 10.1016/j.neunet.2014.09.003. URL: http://dx.doi.org/10.1016/j.neunet.2014.09.003.

[26] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. "Learning representations by back-propagating errors". In: *nature* 323.6088 (1986), pp. 533–536.