UNIPS: UNIFIED PATCH SIMILARITY UNLOCKS TRAINING-FREE VISUAL PLACE RECOGNITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual Place Recognition (VPR) localises an image by retrieving the geo-referenced database image that shows the same place. Most VPR methods are training-based, aiming to learn effective features from annotated data. These methods achieve strong results on standard VPR benchmarks but require millions of labelled samples, making annotation cost a significant challenge. Even with such extensive training, they still fail on other VPR benchmarks that differ from their training set, such as cross-environment benchmarks. Training-free VPR methods avoid this annotation cost and generalise better. However, these methods do not fully leverage pretrained vision models, resulting in features that lack sufficient discriminative power to achieve strong performance on standard VPR benchmarks, highlighting the challenge of effectively utilising pretrained vision models. To tackle these challenges, we propose UniPS (Unified Patch Similarity), a training-free pipeline built solely on patch similarity. Given a query or database image, only a frozen backbone is applied to extract patch features. We then perform a series of patch similarity-based comparisons for VPR: self-similarity for patch importance, followed by weighted second-moment patch feature pooling for coarse ranking, and cross-image patch similarity for reranking. Based on comprehensive evaluation on **17** benchmarks, UniPS achieves state-of-the-art results on both the 8 standard and 9 cross-environment VPR benchmarks. On standard VPR benchmarks, UniPS outperforms the best training-free baseline by **+12.9%** and the best trained baseline by **+0.2%**, despite the latter's heavy annotation requirements. On cross-environment benchmarks, UniPS outperforms the best training-free baseline by **+7.1%** and the best trained baseline by **+11.5%**. In addition, the proposed module can be easily plugged into existing methods, improving their performance by **+5.6%** to up to **+17.8%**.

1 INTRODUCTION

Visual Place Recognition (VPR) is a long-standing computer vision challenge that aims to recognise where a query image was captured by matching it to a database of images tagged with their locations. VPR is essential to many core applications, such as visual SLAM (Taketomi et al., 2017), relocalisation in autonomous driving (Gadd et al., 2021), and city-scale mapping services (Pfaff et al., 2007). Classical and standard VPR benchmarks such as Torii et al. (2013), mainly target this setting, using largely standard street-view images. However, we notice that VPR is increasingly required in a broader range of scenarios (Keetha et al., 2023), such as aerial and drone navigation (Rezwan & Choi, 2022), indoor delivery and service robots (Chung et al., 2007), underwater vehicles surveying the sea floor (Yoerger et al., 2007), and the matching of historical photographs to present-day scenes (Munteán, 2015). We refer to these scenarios and their corresponding benchmarks as cross-environment benchmarks. In this paper, we argue that advanced VPR solutions should generalise not only to standard VPR benchmarks but also to cross-environment benchmarks.

Existing VPR methods are mainly training-based approaches, which require a large amount of annotated data for training. Classical methods such as NetVLAD (Arandjelovic et al., 2016) make soft assignment to a codebook differentiable and train it under weak supervision, while later work follows this template through classification over geographic cells (Berton et al., 2022; 2023), curated labels at scale (Ali-Bey et al., 2022), or feed-forward pooling (Ali-Bey et al., 2023), among others. Recent advanced methods achieve large performance improvements, but these gains are largely driven by increasingly large-scale annotated training sets: for example, GSV-Cities (Ali-Bey et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

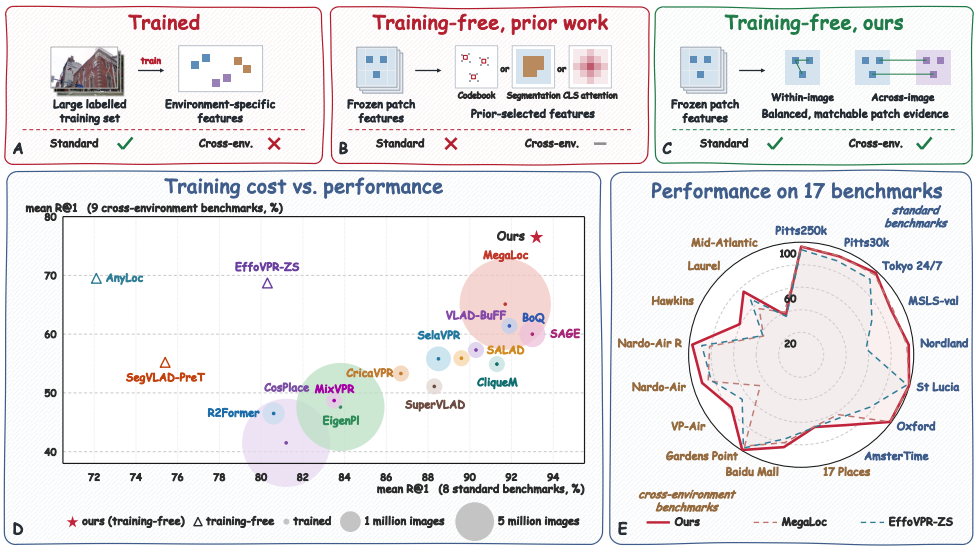

Figure 1: **Overview and performance (R@1).** (A) Trained methods require large-scale labelled datasets, limiting their generalisation to cross-environment settings. (B) Prior training-free methods avoid this training cost and generalise better, but suffer a performance drop on standard VPR benchmarks. (C) Our training-free method instead applies frozen patch features directly, using patch similarity both within and across images for retrieval. (D) Training cost versus performance of our method and baselines on 17 benchmarks, averaged separately across standard and cross-environment VPR benchmarks. (E) Performance of the proposed method against the state-of-the-art trained and training-free baselines.

2022) provides about half a million street-level images over sixty thousand labelled places, while the SF-XL dataset (Berton et al., 2022; 2023) contains more than forty million annotated samples. Recent state-of-the-art methods, such as (Berton & Masone, 2025), gather even more annotated samples to train their models, achieving the best results but relying heavily on the scale of the annotated training set, as shown in Fig. 1D. Even with millions of annotated samples, however, these state-of-the-art methods tend to learn environment- or domain-specific features that do not generalise well as shown in Fig. 1A, suffering a performance drop on cross-environment benchmarks, as shown in Fig. 1D&E.

To address this data-dependency challenge, training-free VPR methods have recently been proposed (Keetha et al., 2023; Tzachor et al., 2025), which aim to exploit the generalisation ability of large vision models, such as DINO backbones (Oquab et al., 2024) or SAM (Kirillov et al., 2023), to better solve VPR across a wide range of environments. These methods often generalise better than trained methods in the *cross-environment* (indoor, day–night, aerial, subterranean and underwater imagery (Sahdev & Tsotsos, 2016; Sünderhauf et al., 2015; Schleiss et al., 2022; Keetha et al., 2023; Boittiaux et al., 2023)) setting, but fail on standard VPR benchmarks, as shown in Fig. 1D-E. We argue that current training-free methods, which apply various strategies to obtain prior-selected features (Fig. 1B), do not fully utilise the potential of these large vision models.

To address these challenges, we propose UniPS (Unified Patch Similarity), a simple and effective training-free method for VPR. The key intuition behind UniPS is that the most useful visual information for place recognition should be determined by whether it can support image matching, rather than selected in advance. Instead of learning a task-specific representation or relying on a fixed choice of salient features, UniPS uses patch similarity itself to identify distinctive evidence within each image and to measure correspondence across images. We use this same signal at different levels of the retrieval process: within an image to suppress repetitive content, across all database images to form a compact representation for scalable retrieval, and finally between patches of matched candidates to verify local correspondence. In this way, UniPS retains the efficiency of global retrieval while bringing the matching specificity of local features into a training-free framework.

- We introduce UniPS, a unified patch-similarity framework for training-free VPR. The method converts patch-level similarity into complementary signals for patch importance,

scalable global retrieval, and local candidate verification, without model training or annotated data.

- Through comprehensive evaluation on 17 benchmarks, we show that UniPS achieves state-of-the-art results on both standard and cross-environment VPR benchmarks, outperforming the best training-free baselines by **+12.9%** and **+7.1%** respectively, and even surpassing the best trained baselines, which rely on tens of millions of annotated samples.
- We show that the proposed reranking module is broadly applicable: it can be plugged into existing VPR methods to improve their generalisation, boosting their performance by **+5.6%** to **+17.8%** on cross-environment benchmarks.

2 Related Work

Supervised and training-free VPR. Most VPR methods are training-based (Izquierdo & Civera, 2024a; Ali-Bey et al., 2023; Qiu et al., 2024) which learn the representations from geo-info annotated data (Arandjelovic et al., 2016). CosPlace and EigenPlaces (Berton et al., 2022; 2023) train classifiers over geographic cells. SALAD (Izquierdo & Civera, 2024b) and BoQ (Ali-Bey et al., 2024) aggregate the features from pretrained backbones such as DINOv2 (Oquab et al., 2024). SuperVLAD (Lu et al., 2024c) removes cluster-centre subtraction but retains learned assignments. SelaVPR (Lu et al., 2024b) tunes adapters in a frozen backbone. CricaVPR (Lu et al., 2024a) attends across the images in a batch, so the descriptor of one image depends on the others. These methods aggregate local features without place supervision. AnyLoc (Keetha et al., 2023) aggregates frozen DINOv2 patches using VLAD with a per-domain vocabulary with higher accuracy than GeM pooling (Radenović et al., 2018). SegVLAD (Garg et al., 2024) applies SAM (Kirillov et al., 2023) to handle partial overlap, which supports generic and task-specialised encoders. Zero-shot EffoVPR (Tzachor et al., 2025) matches frozen patches selected by class-token attention. Our approach weights patches by within-image repetition for both aggregation and candidate matching.

Feature aggregation, weighting and matching. Most VPR methods pool local features into global descriptors for nearest-neighbour retrieval. Fisher vectors (Perronnin & Dance, 2007; Perronnin et al., 2010) encode descriptors relative to a learned codebook and sum these encodings per codeword, while second-order pooling (Lin et al., 2015; Li et al., 2017) averages descriptor outer products into a bilinear or covariance matrix. Discriminative patch mining (Doersch et al., 2015; 2013) identifies local patterns distinguishing image collections. VPR methods discard features recurring across unrelated places (Knopp et al., 2010), down-weight repeated structures (Torii et al., 2013; Jégou et al., 2009) or score distinctiveness (Arandjelović & Zisserman, 2014), and Gronat et al. (2013) learn identifying features with per-location classifiers. SAGE (Chen et al., 2026) learns residual patch weights from place labels before pooling. Generalized max pooling (Murray & Perronnin, 2014) uses ridge regression to equalize their similarity to the aggregate. Our method weights each patch by the inverse count of similar patches within its image for second-order pooling.

Local matching and reranking. Two-stage VPR reranks top-k candidates by local-feature matching. Patch-NetVLAD (Hausler et al., 2021) matches multi-scale patch descriptors with spatial verification; SelaVPR (Lu et al., 2024b) counts mutual nearest neighbours between adapted features. R2Former (Zhu et al., 2023) learns reranking from feature correlations and spatial information, while Pair-VPR (Hausler & Moghadam, 2025) trains a transformer to classify same-place image pairs. FoL (Wang et al., 2025) restricts matching to learned discriminative regions. EffoVPR (Tzachor et al., 2025) uses class-token attention to select features and counts mutual nearest-neighbour matches above a similarity threshold. Our method uses confidence-gated MaxSim reuses these query weights to sum similarities rather than count matches, without task-specific training.

3 Method

As illustrated in Fig. 2, the proposed method UniPS follows a training-free paradigm and requires neither model training nor annotated data. All steps of the pipeline rely on a single patch-to-patch similarity (PS) strategy, expressed in three different forms. The query and the database images are first partitioned into patches, and their patch features are extracted by a frozen pretrained vision model. Based on these features, each image computes the similarity between its own patches (PS-1), from which the weight of every patch is derived (Sec. 3.2). Second-moment pooling of each image's weighted patch features (PS-2) then yields one descriptor per image, with which the database is

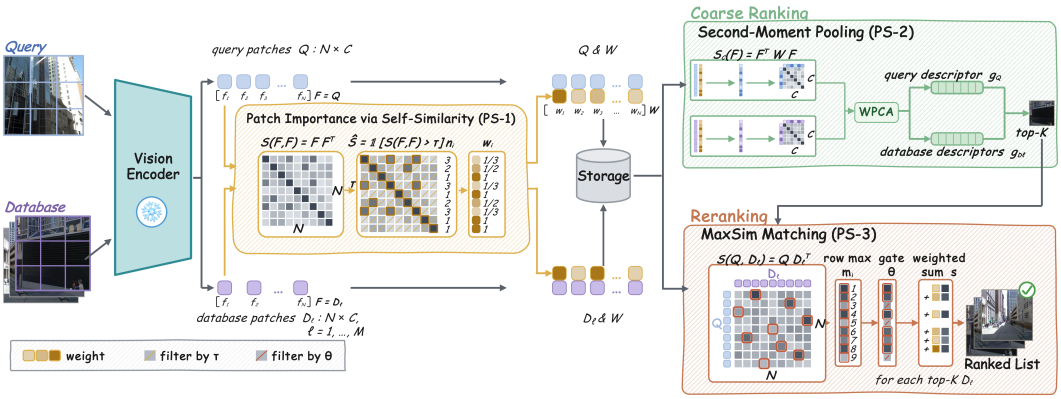

Figure 2: Overview of the proposed method. Each image is partitioned into patches, whose features are extracted by a frozen vision encoder, and the patch similarity (PS) template is applied in three forms. PS-1 computes the self-similarity of each image, which gives the weight of every patch; features and weights are computed once and stored for both stages. PS-2 pools the weighted patch features into a second-moment matrix, reduced by whitened PCA (WPCA) to one descriptor per image, and coarse ranking shortlists the K nearest database images. PS-3 performs weighted MaxSim matching between the query and each shortlisted image, and reranking sorts the shortlist by the matching score.

coarsely ranked, and patch-to-patch MaxSim matching (Barrow et al., 1977) between the query and the top-ranked database images (PS-3) refines this ranking (Sec. 3.3).

3.1 Problem Formulation and Notation

Given a query image I_q and a database $\mathcal{D} = \{(I_d, \mathbf{c}_d)\}$ of M images I_d with their spatial coordinates $\mathbf{c}_d$, VPR retrieves the database image that best matches the query, from which the query's coordinate $\hat{\mathbf{c}}_q$ is inferred (Arandjelovic et al., 2016). As discussed in Sec. 2, most VPR solutions require a massive annotated training set $\mathcal{D}_{\text{train}} = \{(I_t, \mathbf{c}_t)\}$ of training images I_t with coordinates $\mathbf{c}_t$ to train a dedicated retrieval model. Training such a model takes from about six hours for BoQ (Ali-Bey et al., 2024) to about a day for CosPlace and EigenPlaces on the 41M images of SF-XL (Berton et al., 2022; 2023), on up to eight V100 GPUs for R2Former (Zhu et al., 2023), depending on the complexity of the method and the scale of $\mathcal{D}_{\text{train}}$.

The proposed UniPS eliminates this training requirement by performing patch-similarity-based retrieval in a training-free manner. As shown in Fig. 2, prior to retrieval, each input image I (the query I_q or any database image I_d) is resized to a fixed resolution and partitioned into N patches, $I = [p_1, \ldots, p_N]$. A frozen vision encoder extracts a feature vector $\mathbf{f}_i \in \mathbb{R}^C$ for each patch p_i, so that every image is represented by a patch feature matrix $F = [\mathbf{f}_1, \ldots, \mathbf{f}_N]^\top \in \mathbb{R}^{N \times C}$, where C is the feature dimension. Following this notation, we denote the query's patch feature matrix by Q and that of the ℓ-th database image by D_ℓ, $\ell = 1, \ldots, M$. Given these patch features, we adopt a general PS template,

$$S(A, B) = AB^\top, \tag{1}$$

where A and B are set to different feature matrices depending on the form of PS being applied. We first apply this template for patch weighting, and subsequently for coarse ranking and reranking, as detailed below.

3.2 Patch Importance via Self-Similarity (PS-1)

Comparing images through patch correspondences is a long-standing idea (Barnes et al., 2009), and summing each patch's nearest-neighbour similarity already gives a strong image distance without any learning (Boiman et al., 2008). What such a comparison does not say is how much each patch should count: a patch whose appearance recurs many times within its own image cannot tell one place from another.

In this paper, we capture this behaviour through a simple and effective patch-to-patch self-similarity formulation (PS-1). Down-weighting content that recurs within an image goes back to the treatment of repetitive structures in place recognition by Torii et al. (2013), which keeps a few repeated elements from dominating the match. Given the patch feature matrix $F \in \mathbb{R}^{N \times C}$ of an

image, the PS template with $A = B = F$ gives the self-similarity $S(F, F) = FF^\top \in \mathbb{R}^{N\times N}$, whose element $S(F, F)_{ij}$, $i, j \in \{1, \dots, N\}$, is the similarity between patches p_i and p_j. To count how many patches of the same image are highly similar to each patch, we introduce a similarity threshold τ, fixed at 0.5 for all benchmarks (Appendix A.1), and binarise this table, i.e. $\hat{S}_{ij} = \mathbb{1}[S(F, F)_{ij} > \tau]$, where $\mathbb{1}[\cdot]$ is the indicator function. The weight of patch p_i is then defined as

$$w_i = \frac{1}{n_i}, \qquad \text{where} \quad n_i = \sum_{j=1}^{N} \hat{S}_{ij}, \tag{2}$$

and n_i counts the patches, including p_i itself, that are highly similar to p_i, that is, the nonzero entries remaining in row i of $\hat{S}$ after thresholding. This discounts bursty content, visual elements that repeat within a single image and would otherwise dominate any similarity computed from them (Jégou et al., 2009). Since w_i is the reciprocal of n_i, a patch that recurs frequently across the image (large n_i) receives a small weight, while a distinctive patch with few or no similar patches (small n_i) receives a large weight, up to w_i=1, reflecting its higher importance (Fig. 3). The weights of an image are collected in $\mathbf{w} = [w_1, \dots, w_N]$ and applied in the coarse ranking and reranking.

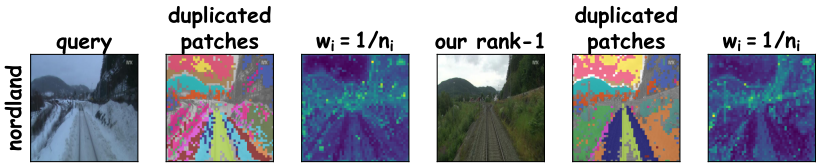

Figure 3: Patch self-similarity weighting on a Nordland pair (query left, our rank-1 retrieval right). For each image: colour clusters indicate similar patches, and grey patches are more distinctive in the image. The heat maps show the resulting weight $w_i = 1/n_i$. Repetitive patches such as snow, sky and track falls into large clusters and receives low weight, while sparse, distinctive content receives high weight. More results are in Appendix C.2.

3.3 Coarse Ranking and Reranking

Two-stage retrieval is standard in VPR (Hausler et al., 2021; Zhu et al., 2023; Lu et al., 2024b), but existing pipelines treat each stage separately. Here both stages use the same frozen tokens with the same weights $\mathbf{w}$: coarse ranking pools them into one descriptor per image (PS-2), and reranking matches them patch by patch on the top-K shortlist (PS-3).

Coarse Ranking (PS-2). During coarse ranking, the query is compared with the whole database $\mathcal{D}$ to obtain the closest top-K candidates, which requires a compact descriptor for every image. We obtain this descriptor by weighted second-moment pooling of the patch features, the bilinear pooling of Lin et al. (2015), which captures pairwise correlations between feature channels. Given the patch feature matrix F of an image and its patch weights $\mathbf{w}$ from PS-1, Eq. (2), let $W = \mathrm{diag}(\mathbf{w})$. Setting $A = B = (W^{1/2}F)^\top$ in Eq. (1), that is, applying the PS template along the channel dimension of the weighted patch features, gives the $C \times C$ second-moment matrix of the image, denoted $S_C(F)$,

$$S_C(F) = S\big((W^{1/2}F)^\top, (W^{1/2}F)^\top\big) = F^\top W F = \sum_{i=1}^{N} w_i\, \mathbf{f}_i \mathbf{f}_i^\top. \tag{3}$$

As S_C is high-dimensional, we apply ℓ_2 normalisation and whitened PCA, denoted ρ, to obtain a compact descriptor $\mathbf{g} = \rho(S_C)$ for efficient large-scale search. Denoting the query and the ℓ-th database descriptors by $\mathbf{g}_Q$ and $\mathbf{g}_{D_\ell}$, coarse ranking sorts all database images by $\mathbf{g}_Q^\top \mathbf{g}_{D_\ell}$, and the top-$K$ images are passed to reranking.

Reranking (PS-3). Reranking is performed between the query Q and each of the K shortlisted database images D_ℓ (Sec. 3.1) via patch-to-patch MaxSim matching (Barrow et al., 1977). Setting $A = Q$ and $B = D_\ell$ in Eq. (1) gives the cross-image similarity $S(Q, D_\ell) \in \mathbb{R}^{N\times N}$, whose entry $S(Q, D_\ell)_{ij}$ is the similarity between query patch p_i and database patch p_j. Let $m_i = \max_j S(Q, D_\ell)_{ij}$ be the best match of query patch p_i, $w_{Q,i}$ its weight from PS-1, and θ a similarity threshold, fixed at 0.65 for all benchmarks (Appendix A.1): a query patch whose best

match falls below θ has no counterpart in the candidate and is discarded. The reranking score $s(Q, D_\ell)$ is defined as

$$s(Q, D_\ell) = \sum_{i=1}^{N} w_{Q,i} \; \mathbb{1}[\, m_i > \theta \,] \; m_i, \tag{4}$$

which sums the weighted best matches of the query patches. The K candidates are sorted by $s(Q, D_\ell)$, and the top-ranked one gives the location of the query. All implementation details of the proposed method are provided in the Appendix A.2.

4 Experiments

4.1 Experimental setup

Table 1: The 17 benchmarks with their data type.

Standard VPR benchmarks		*Cross-environment VPR benchmarks*	
Data types	Benchmarks	Data types	Benchmarks
Viewpoint	Pitts30k, Pitts250k	Indoor	17 Places, Baidu Mall
Illumination	Tokyo 24/7		Gardens Point
Season, weather	MSLS-val	Aerial	VP-Air, Nardo-Air
Railway, seasons	Nordland		Nardo-Air R (rotated)
Route repetition	St Lucia, Oxford RobotCar	Subterranean	Hawkins, Laurel Caverns
Long-term change	AmsterTime	Underwater	Mid-Atlantic Ridge

Datasets. In this paper, we evaluate the proposed method against a wide range of trained and training-free baselines on **17** benchmarks. Table 1 summarises these benchmarks in two groups, along with their data type. The first group includes 8 standard VPR benchmarks: 7 street-level ones, Pitts250k and Pitts30k (Torii et al., 2013), Tokyo 24/7 (Torii et al., 2015), MSLS-val (Warburg et al., 2020), St Lucia (Milford & Wyeth, 2008), Oxford RobotCar (Maddern et al., 2017) and AmsterTime (Yildiz et al., 2022), together with the Nordland railway benchmark (Sünderhauf et al., 2013). The second group consists of 9 cross-environment localisation benchmarks: 17 Places (Sahdev & Tsotsos, 2016) and Baidu Mall (Sun et al., 2017) indoors, Gardens Point (Sünderhauf et al., 2015) across day and night, VP-Air (Schleiss et al., 2022) and Nardo-Air with its rotated variant in the air, Hawkins and Laurel Caverns underground, and the Mid-Atlantic Ridge (Boittiaux et al., 2023) underwater, following the setup of Keetha et al. (2023).

Evaluation metrics. We report Recall@1 (R@1) under the standard protocol (Arandjelovic et al., 2016; Hausler et al., 2021): a query counts as localised if the top-ranked database image lies within the tolerance of its benchmark, following the order of Table 1. For the standard benchmarks, this is 25 m, except Nordland (± 10 frames), Oxford RobotCar (16 m) and AmsterTime (the paired present-day image); for the cross-environment benchmarks, it is the provided correspondences or pose thresholds.

4.2 Comparison with state-of-the-art

We compare the proposed UniPS with a comprehensive set of baselines under both trained and training-free settings. On average, UniPS achieves state-of-the-art results on both Standard VPR benchmarks and Cross-environment VPR benchmarks, against both types of baselines. Note that, UniPS achieves the best results on many standard VPR benchmarks against trained baselines, which require millions of annotated samples. Detailed comparisons against training-free and trained baselines are presented below.

Against training-free methods. We first compare UniPS against existing training-free baselines on all 17 benchmarks: the pooling baselines, the frozen class token (Oquab et al., 2024) and GeM (Radenović et al., 2018), and the more recent AnyLoc (Keetha et al., 2023), SegVLAD-PreT (Garg et al., 2024) and EffoVPR-ZS (Tzachor et al., 2025). As shown in Table 2, on the standard benchmarks, UniPS achieves a mean of 93.2 against 80.3 for EffoVPR-ZS. The largest improvement is on the Nordland benchmark, at $+38.1$, where most training-free methods fail. On the cross-environment benchmarks, UniPS again achieves the best results with average performance 76.6 over 69.5 for AnyLoc, whose descriptor is $24\times$ larger than ours. Over all 17 benchmarks, UniPS

achieves 84.4, +10.3 over the best prior training-free method. The full version of Table 2 is included in Appendix B.1.

Table 2: Comparison against training-free baselines. R@1 (%) on the 4 standard and 4 cross-environment benchmarks, with the mean over all 8 standard benchmarks, all 9 cross-environment benchmarks, and all 17 benchmarks. Dim is the dimension of the global descriptor of each method. † indicates two-stage methods. Best and second-best results are in **Bold** and Underline. Ours_{D2} and Ours_{D3} use a frozen DINOv2 (Oquab et al., 2024) and DINOv3 (Siméoni et al., 2025) backbone.

		Standard VPR benchmarks					Cross-environment VPR benchmarks					
Method	Dim	Pitts30k	Tokyo24/7	Nordland	AmsterTime	mean (8)	Hawk	Laurel	Baidu	VP-Air	mean (9)	mean (17)
CLS token	1024	78.7	46.0	15.9	45.8	51.5	26.3	50.9	43.1	38.0	50.2	50.8
GeM	1024	78.5	69.8	15.0	27.9	56.7	56.8	50.9	46.2	42.2	60.1	58.5
AnyLoc	49152	87.7	82.9	23.9	37.8	72.1	65.2	61.6	75.2	66.7	69.5	70.8
EffoVPR-ZS†	1024	89.4	90.8	57.9	65.8	80.3	38.1	67.0	75.9	65.7	68.7	74.1
SegVLAD-PreT	49152	86.7	94.3	26.3	62.4	75.4	56.8	26.8	78.5	69.8	55.2	64.7
Ours_{D2}†	1024	92.5	97.1	82.3	68.2	90.3	**67.0**	**83.0**	**84.5**	72.2	76.3	82.9
Ours_{D3}†	1024	**93.7**	**98.4**	**96.0**	**73.6**	**93.2**	61.0	75.9	83.5	**77.7**	**76.6**	**84.4**

Against trained methods. Table 3 compares UniPS against trained methods on all 17 benchmarks, from the earlier NetVLAD (Arandjelovic et al., 2016), SFRS (Ge et al., 2020), CosPlace (Berton et al., 2022), MixVPR (Ali-Bey et al., 2023), EigenPlaces (Berton et al., 2023) and R2Former (Zhu et al., 2023) to the recent SALAD (Izquierdo & Civera, 2024b), CricaVPR (Lu et al., 2024a), SelaVPR (Lu et al., 2024b), BoQ (Ali-Bey et al., 2024), CliqueMining (Izquierdo & Civera, 2024a), SuperVLAD (Lu et al., 2024c), VLAD-BuFF (Khaliq et al., 2024), MegaLoc (Berton & Masone, 2025) and SAGE (Chen et al., 2026). On the standard benchmarks, UniPS achieves a mean R@1 of **93.2**, surpassing the current state-of-the-art method, SAGE, at 93.0, with the best R@1 on Tokyo24/7, Nordland, and AmsterTime. On the cross-environment benchmarks, all trained methods suffer a significant performance drop, e.g., SAGE falls from 93.0 to 60.0 mean R@1, whereas UniPS drops by only 16.6 at R@1. On average, UniPS outperforms the state-of-the-art trained method, MegaLoc, with **76.6** against 65.1. Averaged over all 17 benchmarks, UniPS outperforms the second-best method, MegaLoc, by **6.8** at R@1. Notably, UniPS is training-free and requires no annotated data, while all trained methods rely on large-scale annotated training datasets. For example, as shown in Fig. 1, EigenPlaces is trained on 41.2 million images, and MegaLoc needs 45 million to reach its state-of-the-art results.

Table 3: Comparison against trained methods.R@1 (%) on the 4 standard and 4 cross-environment benchmarks, with the mean over all 8 standard benchmarks, all 9 cross-environment benchmarks, and all 17 benchmarks. The full table is presented in Appendix B.1. † indicates two-stage methods; ‡ indicates inductive setting. Best and second-best results are in **Bold** and Underline. Ours_{D2} and Ours_{D3} use a frozen DINOv2 and DINOv3 backbone.

		Standard VPR benchmarks					Cross-environment VPR benchmarks					
Method	Dim	Pitts30k	Tokyo24/7	Nordland	AmsterTime	mean (8)	Hawk	Laurel	Baidu	VP-Air	mean (9)	mean (17)
NetVLAD	32768	82.1	54.9	6.4	20.6	47.9	38.1	42.9	57.8	14.7	39.9	43.7
SFRS	4096	70.9	85.4	8.8	19.3	52.8	29.7	21.4	51.3	9.4	31.5	41.5
CosPlace	2048	90.9	87.3	63.0	49.7	81.2	33.9	25.9	47.1	10.4	41.5	60.2
MixVPR	4096	91.6	86.3	76.6	40.9	83.5	26.3	32.1	68.4	12.9	48.7	65.1
EigenPlaces	2048	92.5	93.0	63.6	48.9	83.8	28.8	27.7	66.6	13.1	47.6	64.6
R2Former†	256	91.1	88.6	60.5	29.5	80.6	17.8	30.4	74.7	15.0	46.5	62.6
SALAD	8448	92.4	94.6	86.5	58.4	89.6	38.1	46.4	69.9	24.0	55.9	71.8
CricaVPR‡	10752	91.8	90.2	80.5	49.0	86.7	28.0	34.8	65.8	19.9	53.3	69.0
SelaVPR†	1024	92.8	94.0	85.2	50.3	88.5	25.4	53.6	68.5	32.6	55.8	71.1
BoQ	12288	93.7	98.1	90.6	63.0	91.9	33.9	60.7	77.4	29.5	61.4	75.8
CliqueMining	8448	92.6	96.8	95.9	57.7	91.3	22.9	38.4	69.1	22.9	54.9	72.0
SuperVLAD‡	3072	92.6	93.7	79.2	54.7	88.3	37.3	51.8	70.3	23.8	51.1	68.6
VLAD-BuFF	4096	92.5	96.5	84.9	61.4	90.3	37.3	33.9	76.4	33.3	57.3	72.8
MegaLoc	8448	94.1	96.5	94.3	62.5	91.7	37.3	55.4	79.4	45.6	65.1	77.6
SAGE‡	8448	**94.7**	97.8	95.1	66.0	93.0	27.1	46.4	74.6	39.1	60.0	75.5
Ours_{D2}†	1024	92.5	97.1	82.3	68.2	90.3	**67.0**	**83.0**	**84.5**	72.2	76.3	82.9
Ours_{D3}†	1024	93.7	**98.4**	**96.0**	**73.6**	**93.2**	61.0	75.9	83.5	**77.7**	**76.6**	**84.4**

Qualitative results. Figure 4 compares UniPS with other baselines on samples across different benchmarks. The state-of-the-art trained methods can also handle hard samples within the standard

benchmarks, but fail beyond them. For example, SAGE correctly retrieves the right place under seasonal change on Nordland, and MegaLoc succeeds on AmsterTime, but both fail in the degraded Hawkins corridor. The state-of-the-art training-free method shows the opposite pattern: AnyLoc succeeds on Hawkins but fails on the 3 standard benchmarks. The proposed UniPS, however, is robust across all cases. More qualitative results are presented in Appendix C.1.

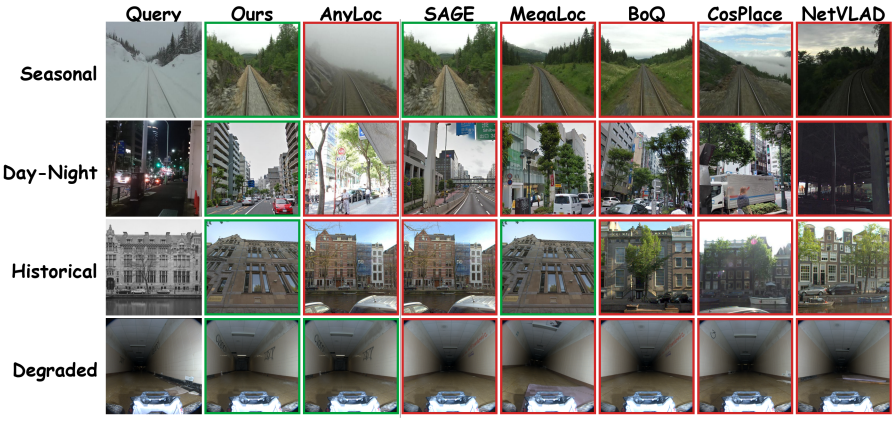

Figure 4: Rank-1 retrievals on 4 benchmarks among the proposed method and other baselines. Green indicates a correct result, and red ones are incorrect results.

4.3 Generalisation of the proposed re-ranking

In addition to achieving state-of-the-art performance on its own, the proposed re-ranking module can be easily plugged into other baselines to improve their generalisation. For two-stage methods such as SelaVPR and R2Former, we simply replace their re-ranker with our patch similarity-based re-ranking, keeping the rest of the pipeline unchanged. As shown in Table 4, both methods improve their mean R@1 over the 9 cross-environment benchmarks, by +9.7% and +5.6%, respectively. For methods with no re-ranking stage (single-stage methods), our re-ranking module can be attached to the original pipeline, which is modified to return top-K candidates instead of only the top-1 result. With our re-ranking module attached, 8 single-stage methods improve their generalisation, with mean R@1 over the 9 cross-environment benchmarks increasing by +9.1% for MegaLoc to up to +17.8% for SuperVLAD. In particular, none of these improvements requires additional annotated data or further training.

Table 4: Robustness of the proposed reranking module. The module is plugged into different baselines and tested on 9 cross-environment benchmarks. Apply our methods on both two-stage and one stage methods. All methods are with their default settings. ‡ indicates inductive setting.

	Degr.	SubT		Indoor			Aerial			UW	
Method	Hawk	Laurel	17Pl	Gard	Baidu	N-Air	N-AirR	VP-Air	M-Atl	mean
Two-stage methods											
SelaVPR	$25.4_{+38.2}$	$53.6_{+13.4}$	$63.3_{+2.5}$	$94.5_{+1.5}$	$68.5_{+7.5}$	$59.1_{+1.5}$	$83.1_{+1.4}$	$32.6_{+23.5}$	$21.8_{-2.0}$	$55.8_{+9.7}$
R2Former	$17.8_{+22.9}$	$30.4_{+16.1}$	$63.8_{+0.7}$	$84.0_{-3.5}$	$74.7_{+0.6}$	$26.8_{+5.6}$	$83.1_{-1.4}$	$15.0_{+2.8}$	$22.8_{+6.9}$	$46.5_{+5.6}$
One-stage methods											
SALAD	$38.1_{+24.6}$	$46.4_{+25.9}$	$64.3_{+1.0}$	$96.5_{+3.0}$	$69.9_{+12.1}$	$53.5_{+29.6}$	$94.4_{+1.4}$	$24.0_{+22.1}$	$15.8_{+14.9}$	$55.9_{+14.9}$
CricaVPR‡	$28.0_{+31.4}$	$34.8_{+32.1}$	$64.0_{+1.5}$	$94.0_{+4.5}$	$65.8_{+18.9}$	$45.1_{+18.3}$	$100.0_{-8.5}$	$19.9_{+22.8}$	$27.7_{+8.9}$	$53.3_{+14.4}$
BoQ	$33.9_{+33.1}$	$60.7_{+16.1}$	$64.3_{+0.7}$	$98.0_{+1.0}$	$77.4_{+6.5}$	$62.0_{+31.0}$	$91.5_{-8.5}$	$29.5_{+24.9}$	$35.6_{-5.0}$	$61.4_{+11.1}$
CliqueMining	$22.9_{+39.8}$	$38.4_{+35.7}$	$65.3_{-0.2}$	$93.0_{+6.0}$	$69.1_{+13.8}$	$62.0_{+8.5}$	$84.5_{+11.3}$	$22.9_{+26.2}$	$35.6_{-1.0}$	$54.9_{+15.6}$
SuperVLAD‡	$37.3_{+23.7}$	$51.8_{+21.4}$	$64.8_{+1.5}$	$95.5_{+3.5}$	$70.3_{+15.1}$	$14.1_{+60.6}$	$88.7_{-9.9}$	$23.8_{+23.4}$	$13.9_{+20.8}$	$51.1_{+17.8}$
MegaLoc	$37.3_{+33.1}$	$55.4_{+17.0}$	$64.8_{+1.2}$	$95.5_{+3.0}$	$79.4_{+5.7}$	$85.9_{+1.4}$	$81.7_{+9.9}$	$45.6_{+16.3}$	$40.6_{-5.9}$	$65.1_{+9.1}$
SAGE‡	$27.1_{+37.3}$	$46.4_{+26.8}$	$64.0_{+0.5}$	$93.5_{+5.0}$	$74.6_{+6.7}$	$85.9_{-9.9}$	$84.5_{+5.6}$	$39.1_{+26.3}$	$24.8_{-2.0}$	$60.0_{+10.7}$
VLAD-BuFF	$37.3_{+23.7}$	$33.9_{+33.0}$	$64.5_{+1.0}$	$96.5_{+3.0}$	$76.4_{+8.1}$	$57.7_{+9.9}$	$88.7_{+5.6}$	$33.3_{+20.1}$	$27.7_{+7.9}$	$57.3_{+12.5}$

4.4 Ablation study

Contribution of each component. We conduct an ablation study to examine the contribution of each component in UniPS: patch weighting (PS1), coarse ranking (PS2), and re-ranking (PS3). Starting from an unweighted second-moment pooling baseline, we add each component in turn,

and report the resulting R@1 on 2 example benchmarks, Nordland (standard) and VP-Air (cross-environment), in Fig. 5. Patch weighting (PS1) provides the largest gain on Nordland (+34.9%), while MAXSIM re-ranking (PS3) provides the largest gain on VP-Air (+12.1%), and adding patch weighting to the re-ranking stage further improves both benchmarks (+6.2% and +10.8%, respectively). These results show that every component of UniPS contributes to the final performance, though their relative importance differs between standard and cross-environment settings.

Table 5: Alternatives to obtain the patch importance: mean R@1 over the 8 standard (Std.), the 9 cross-environment and all 17 benchmarks. Per-benchmark values and Recall@5 and 10 are in Appendix B.3.

Patch importance methods	Std. (8)	Cross (9)	All
MAXSIM (Barrow et al., 1977)	81.6	74.3	77.7
rapid spatial verification (Hausler et al., 2021)	82.7	63.1	72.3
cross-referencing (Zhang et al., 2020)	37.5	44.6	41.2
inverse document frequency (Sivic & Zisserman, 2003)	69.9	73.8	72.0
token norm (Darcet et al., 2024)	82.9	75.0	78.7
class-token attention (Tzachor et al., 2025)	87.8	71.7	79.3
patch-graph centrality (Liu et al., 2026)	88.2	75.4	81.4
self-similarity (ours)	**90.3**	**76.3**	**82.9**

Patch importance alternatives. We test alternative ways to obtain the patch weights w. Table 5 reports the mean results across all benchmark groups. Our self-similarity signal leads in both groups, reaching 90.3% and 76.3% mean R@1, an improvement of +8.7% and +2.0% over unweighted MAXSIM, and outperforming patch-graph centrality (88.2%, 75.4%) and class-token attention (87.8%, 71.7%). Beyond accuracy, self-similarity is also simpler to obtain: it requires only the patch features themselves, whereas the alternatives depend on an attention map from a dedicated network layer.

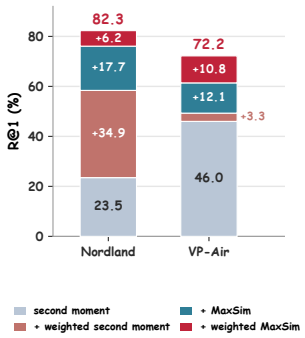

Figure 5: Contribution of each component: Performance at R@1 when each component added.

Same-backbone comparison. As most of the advanced baselines are built on DINOv2, we retrain three representative baselines, SALAD, BoQ, and CliqueMining, on DINOv3 using their released implementations and compare them against UniPS under the same backbone. Table 6 reports per-benchmark results, with a graph visualisation in Appendix B.2. UniPS achieve a mean R@1 of 84.4% across all 17 benchmarks, outperforming BoQ, the best retrained baseline, at 77.3%. Combined with the DINOv2 results in Table 3, this shows that UniPS outperforms all tested baselines with the same backbone, while requiring no annotated data and using at least 8× fewer feature dimensions.

Table 6: Same backbone (DINOv3) tests across 17 benchmarks including three trained methods and the proposed UniPS. Best and second-best results are in **Bold** and Underline.

Method	Dim	P250k	P30k	Tokyo	MSLS	Nord	StLuc	Oxf	Amst	Baidu	17Pl	Gard	Hawk	Laurel	N-Air	N-AirR	VP-Air	M-Atl	*mean*
SALAD	8448	96.2	93.3	97.8	89.0	91.0	99.9	94.2	64.3	64.4	62.3	95.5	44.1	61.6	78.9	87.3	42.6	19.8	75.4
BoQ	12288	**96.8**	**94.4**	**98.7**	91.9	94.3	**100.0**	**99.0**	68.2	71.7	63.8	98.5	33.0	48.2	84.5	**100.0**	49.8	21.8	77.3
CliqueMining	8448	96.1	93.6	98.1	**93.5**	94.8	**100.0**	**99.0**	60.4	68.6	61.6	95.0	22.0	37.5	76.1	90.1	47.2	13.9	73.4
Ours	1024	96.1	93.7	98.4	89.3	**96.0**	99.7	**99.0**	**73.6**	**83.5**	**65.3**	**99.5**	**61.0**	**75.9**	**91.5**	97.2	**77.7**	**37.6**	**84.4**

5 CONCLUSION

We introduced UniPS, a training-free pipeline for VPR that relies purely on different forms of patch similarity at each stage and needs no labelled data. Through comprehensive evaluation on 17 benchmarks, spanning both standard and cross-environment settings, we show that UniPS achieves state-of-the-art results against both training-free methods and trained methods that require millions of annotated images. Beyond the full pipeline, the re-ranking component alone proves effective as a standalone addition to existing VPR methods.

AI USE STATEMENT

In this work, Large Language Models (LLMs) were used to aid in the polishing of the manuscript. Specifically, we used an LLM to assist in refining the language, improving readability, and ensuring clarity in various sections of the paper. The model helped with tasks such as sentence rephrasing, grammar checking, and enhancing the overall flow of the text.

It is important to note that the LLM was not involved in the ideation, research methodology, or experimental design. All research concepts, ideas, and analyses were developed and conducted by the authors. The contributions of the LLM were solely focused on improving the linguistic quality of the paper, with no involvement in the scientific content or data analysis.

The authors take full responsibility for the content of the manuscript, including any text generated or polished by the LLM. We have ensured that the LLM-generated text adheres to ethical guidelines and does not contribute to plagiarism or scientific misconduct.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. In this study, no animal experimentation was involved. All datasets used were sourced in compliance with relevant usage guidelines, ensuring no violation of privacy. We have taken care to avoid any biases or discriminatory outcomes in our research process. No personally identifiable information was used, and no experiments were conducted that could raise privacy or security concerns. We are committed to maintaining transparency and integrity throughout the research process.

REPRODUCIBILITY STATEMENT

We have made every effort to ensure that the results presented in this paper are reproducible. All code will be made publicly available to facilitate replication and verification after the paper gets accepted. The experimental setup, including model configurations, resources, and hardware details, is described in detail in the paper. Additionally, the datasets used in this paper are publicly available, ensuring consistent and reproducible evaluation results. We believe these measures will enable other researchers to reproduce our work and further advance the field.

REFERENCES

Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 513:194–203, 2022.

Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. Mixvpr: Feature mixing for visual place recognition. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2997–3006, 2023.

Amar Ali-Bey, Brahim Chaib-Draa, and Philippe Giguere. Boq: A place is worth a bag of learnable queries. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17794–17803. IEEE, 2024.

Relja Arandjelović and Andrew Zisserman. Dislocation: Scalable descriptor distinctiveness for location recognition. In *Asian conference on computer vision*, pp. 188–204. Springer, 2014.

Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5297–5307, 2016.

Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.*, 28(3):24, 2009.

Harry G. Barrow, Jay M. Tenenbaum, Robert C. Bolles, and Helen C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, pp. 659–663, 1977.

Gabriele Berton and Carlo Masone. Megaloc: One retrieval to place them all. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2852–2858. IEEE, 2025.

Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geo-localization for large-scale applications. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4868–4878. IEEE, 2022.

Gabriele Berton, Gabriele Trivigno, Barbara Caputo, and Carlo Masone. Eigenplaces: Training viewpoint robust models for visual place recognition. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11046–11056. IEEE, 2023.

Oren Boiman, Eli Shechtman, and Michal Irani. In defense of nearest-neighbor based image classification. In *2008 IEEE conference on computer vision and pattern recognition*, pp. 1–8. Ieee, 2008.

Clémentin Boittiaux, Claire Dune, Maxime Ferrera, Aurélien Arnaubec, Ricard Marxer, Marjolaine Matabos, Loïc Van Audenhaege, and Vincent Hugel. Eiffel tower: A deep-sea underwater dataset for long-term visual localization. *The International Journal of Robotics Research*, 42(9):689–699, 2023.

Shunpeng Chen, Changwei Wang, Rongtao Xu, Xingtian Pei, Yukun Song, Jinzhou Lin, Wenhao Xu, Jingyi Zhang, Li Guo, and Shibiao Xu. SAGE: Spatial-visual adaptive graph exploration for efficient visual place recognition. In *The Fourteenth International Conference on Learning Representations*, 2026.

Woojin Chung, Gunhee Kim, and Munsang Kim. Development of the multi-functional indoor service robot psr systems. *Autonomous Robots*, 22(1):1–17, 2007.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International conference on learning representations*, volume 2024, pp. 2632–2652, 2024.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. Mid-level visual element discovery as discriminative mode seeking. *Advances in neural information processing systems*, 26, 2013.

Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei A Efros. What makes paris look like paris? *Communications of the ACM*, 58(12):103–110, 2015.

Matthew Gadd, Daniele De Martini, and Paul Newman. Contrastive learning for unsupervised radar place recognition. In *2021 20th International Conference on Advanced Robotics (ICAR)*, pp. 344–349. IEEE, 2021.

Kartik Garg, Sai Shubodh Puligilla, Shishir Kolathaya, Madhava Krishna, and Sourav Garg. Revisit anything: Visual place recognition via image segment retrieval. In *European Conference on Computer Vision*, pp. 326–343. Springer, 2024.

Yixiao Ge, Haibo Wang, Feng Zhu, Rui Zhao, and Hongsheng Li. Self-supervising fine-grained region similarities for large-scale image localization. In *European conference on computer vision*, pp. 369–386. Springer, 2020.

Petr Gronat, Guillaume Obozinski, Josef Sivic, and Tomas Pajdla. Learning and calibrating per-location classifiers for visual place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 907–914, 2013.

Stephen Hausler and Peyman Moghadam. Pair-vpr: Place-aware pre-training and contrastive pair classification for visual place recognition with vision transformers. *IEEE Robotics and Automation Letters*, 10(4):4013–4020, 2025.

Stephen Hausler, Sourav Garg, Ming Xu, Michael Milford, and Tobias Fischer. Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In *2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 14136–14147. IEEE, 2021.

Sergio Izquierdo and Javier Civera. Close, but not there: Boosting geographic distance sensitivity in visual place recognition. In *European Conference on Computer Vision*, pp. 240–257. Springer, 2024a.

Sergio Izquierdo and Javier Civera. Optimal transport aggregation for visual place recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17658–17668, 2024b.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. On the burstiness of visual elements. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 1169–1176. IEEE, 2009.

Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*, 9(2):1286–1293, 2023.

Ahmad Khaliq, Ming Xu, Stephen Hausler, Michael Milford, and Sourav Garg. Vlad-buff: Burst-aware fast feature aggregation for visual place recognition. In *European Conference on Computer Vision*, pp. 447–466. Springer, 2024.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3992–4003, 2023.

Jan Knopp, Josef Sivic, and Tomas Pajdla. Avoiding confusing features in place recognition. In *European Conference on Computer Vision*, pp. 748–761. Springer, 2010.

Peihua Li, Jiangtao Xie, Qilong Wang, and Wangmeng Zuo. Is second-order information helpful for large-scale visual recognition? In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2089–2097. IEEE, 2017.

Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pp. 1449–1457, 2015.

Zhuchenyang Liu, Ziyu Hu, Yao Zhang, and Yu Xiao. Structural anchor pruning: Training-free multi-vector compression for visual document retrieval. *arXiv preprint arXiv:2601.20107*, 2026.

Feng Lu, Xiangyuan Lan, Lijun Zhang, Dongmei Jiang, Yaowei Wang, and Chun Yuan. Cricavpr: Cross-image correlation-aware representation learning for visual place recognition. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16772–16782. IEEE, 2024a.

Feng Lu, Lijun Zhang, Xiangyuan Lan, Shuting Dong, Yaowei Wang, and Chun Yuan. Towards seamless adaptation of pre-trained models for visual place recognition. In *International conference on learning representations*, volume 2024, pp. 14314–14335, 2024b.

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. *Advances in Neural Information Processing Systems*, 37:5789–5816, 2024c.

Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The Oxford RobotCar dataset. *The International Journal of Robotics Research*, 36(1):3–15, 2017.

Michael J Milford and Gordon F Wyeth. Mapping a suburb with a single camera using a biologically inspired SLAM system. *IEEE Transactions on Robotics*, 24(5):1038–1053, 2008.

László Munteán. Of time and the city: Urban rephotography and the memory of war. *Observatorio (OBS*)*, 2015. doi: 10.15847/obsOBS002015975.

Naila Murray and Florent Perronnin. Generalized max pooling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2473–2480, 2014.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Florent Perronnin and Christopher Dance. Fisher kernels on visual vocabularies for image categorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. IEEE, 2007.

Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the Fisher kernel for large-scale image classification. In *European Conference on Computer Vision*, pp. 143–156. Springer, 2010.

Patrick Pfaff, Rudolph Triebel, Cyrill Stachniss, Pierre Lamon, Wolfram Burgard, and Roland Siegwart. Towards mapping of cities. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 4807–4813. IEEE, 2007.

Qibo Qiu, Shun Zhang, Haiming Gao, Honghui Yang, Haochao Ying, Wenxiao Wang, and Xiaofei He. Emvp: Embracing visual foundation model for visual place recognition with centroid-free probing. *Advances in Neural Information Processing Systems*, 37:120928–120950, 2024.

Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7):1655–1668, 2018.

Sifat Rezwan and Wooyeol Choi. Artificial intelligence approaches for uav navigation: Recent advances and future challenges. *IEEE access*, 10:26320–26339, 2022.

Raghavender Sahdev and John K Tsotsos. Indoor place recognition system for localization of mobile robots. In *2016 13th Conference on Computer and Robot Vision (CRV)*, pp. 53–60. IEEE, 2016.

Michael Schleiss, Fahmi Rouatbi, and Daniel Cremers. VPAIR – aerial visual place recognition and localization in large-scale outdoor environments. *arXiv preprint arXiv:2205.11567*, 2022.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 1470–1477, 2003.

Xun Sun, Yuanfan Xie, Pei Luo, and Liang Wang. A dataset for benchmarking image-based localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5641–5649, 2017.

Niko Sünderhauf, Peer Neubert, and Peter Protzel. Are we there yet? challenging SeqSLAM on a 3000 km journey across all four seasons. In *Proceedings of the Workshop on Long-Term Autonomy, IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2013, 2013.

Niko Sünderhauf, Sareh Shirazi, Feras Dayoub, Ben Upcroft, and Michael Milford. On the performance of ConvNet features for place recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4297–4304. IEEE, 2015.

Takafumi Taketomi, Hideaki Uchiyama, and Sei Ikeda. Visual slam algorithms: A survey from 2010 to 2016. *IPSJ transactions on computer vision and applications*, 9(1):16, 2017.

Akihiko Torii, Josef Sivic, Tomas Pajdla, and Masatoshi Okutomi. Visual place recognition with repetitive structures. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 883–890, 2013.

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*, pp. 1808–1817, 2015.

Issar Tzachor, Boaz Lerner, Matan Levy, Michael Green, Tal Berkovitz Shalev, Gavriel Habib, Dvir Samuel, Noam Zailer, Or Shimshi, Nir Darshan, et al. Effovpr: Effective foundation model utilization for visual place recognition. In *International Conference on Learning Representations*, volume 2025, pp. 42817–42839, 2025.

Changwei Wang, Shunpeng Chen, Yukun Song, Rongtao Xu, Zherui Zhang, Jiguang Zhang, Haoran Yang, Yu Zhang, Kexue Fu, Shide Du, et al. Focus on local: Finding reliable discriminative regions for visual place recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7536–7544, 2025.

Frederik Warburg, Soren Hauberg, Manuel Lopez-Antequera, Pau Gargallo, Yubin Kuang, and Javier Civera. Mapillary street-level sequences: A dataset for lifelong place recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2626–2635, 2020.

Burak Yildiz, Seyran Khademi, Ronald Maria Siebes, and Jan van Gemert. AmsterTime: A visual place recognition benchmark dataset for severe domain shift. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 2749–2755. IEEE, 2022.

Dana R Yoerger, Michael Jakuba, Albert M Bradley, and Brian Bingham. Techniques for deep sea near bottom survey using an autonomous underwater vehicle. *The International Journal of Robotics Research*, 26(1):41–54, 2007.

Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 12200–12210. IEEE, 2020.

Sijie Zhu, Linjie Yang, Chen Chen, Mubarak Shah, Xiaohui Shen, and Heng Wang. r2 former: Unified retrieval and reranking transformer for place recognition. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19370–19380. IEEE, 2023.

APPENDIX

This appendix provides implementation details and additional evaluations of UniPS. We describe threshold selection, feature extraction, descriptor compression, and per-query runtime in Sec. A. We then present in Sec. B comparisons with training-free and trained methods on all 17 benchmarks, the matched-backbone comparison, component ablations, Recall@1/5/10, and comparisons of patch importance signals. Sec. C visualises duplicated patches and their self-similarity weights for query–retrieval pairs from every benchmark. Finally, we include a section for limitation and future work in Sec. D

A EXPERIMENTAL DETAILS

A.1 THRESHOLD τ AND θ SELECTION

Both thresholds are calibrated based on SF-XL val, which is isloated from the tested benchmarks in this paper. They are fixed for all the tests among 17 benchmarks in this paper.

The gate of Eq. (4) keeps a query patch when its best match m_i exceeds θ: the likelihood that a match is genuine rather than accidental increases with m_i, so a fixed threshold is the Bayes-optimal rule.

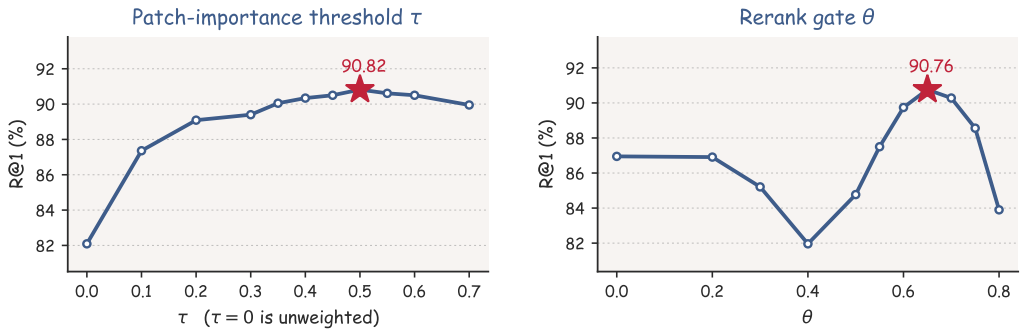

Figure 6: Calibration on SF-XL val. We fix τ=0.5 and θ=0.65, marked by the star.

A.2 IMPLEMENTATION DETAILS

All experiments share one configuration. The backbone is a frozen DINOv3 for our best results and has DINOv2 for the Ours$_{\text{D2}}$ results; its value-facet tokens at 672px (DINOv2) or 768px (DINOv3) are reduced to C=512 channels by PCA. The coarse descriptor S_C is reduced to 1,024 dimensions by whitened PCA. We set τ=0.5 and θ=0.65 for all benchmarks (Sec. A.1).

A.3 PER-QUERY COST IN TIME

We report the processing time (ms) for a single query at every stage of all two-stage methods All of the experiments are on the same hardware.

Table 7: Per-query cost

Method	Extraction	Coarse ranking	Reranking	Total
Patch-NetVLAD	15.5	0.3	5302	5318
R2Former	**4.3**	0.1	62.4	66.8
SelaVPR	14.7	0.1	28.6	43.4
EffoVPR-ZS	15.3	0.1	19.3	**34.7**
Ours$_{\text{D2}}$	31.5	0.2	**15.5**	47.2

B ADDITIONAL RESULTS

In this section, we provide additional quantitative results, which present the full results of the trained and training-free methods across all 17 VPR benchmarks, i.e., both standard benchmarks and cross-environment benchmarks.

B.1 FULL COMPARISON WITH TRAINING-FREE AND TRAINED METHODS

Table 8 includes all the test results for training-free methods, which is the full version of Table 2 in the main paper. Table 9 includes all the test results for training-free methods, which is the full version of Table 3 in the main paper. Both results consistently show that the proposed method UniPS achieves state-of-the-art results.

Table 8: Full version of Table 2

Standard VPR benchmarks Method	Dim	Pitts250k	Pitts30k	Tokyo24/7	MSLS-val	Nordland	St Lucia	Oxford	AmsterTime	mean
Training-free										
CLS token	1024	80.3	78.7	46.0	37.8	15.9	81.5	25.6	45.8	51.5
GeM	1024	80.4	78.5	69.8	36.0	15.0	74.2	71.7	27.9	56.7
AnyLoc	49152	87.1	87.7	82.9	61.9	23.9	96.2	**99.5**	37.8	72.1
EffoVPR-ZS†	1024	93.4	89.4	90.8	70.3	57.9	97.8	77.0	65.8	80.3
SegVLAD-PreT	49152	86.0	86.7	94.3	76.6	26.3	97.1	73.8	62.4	75.4
Ours_{D2}†	1024	95.7	92.5	97.1	89.0	82.3	99.0	99.0	68.2	90.3
Ours_{D3}†	1024	**96.1**	**93.7**	**98.4**	**89.3**	**96.0**	**99.7**	99.0	**73.6**	**93.2**

Cross-environment VPR benchmarks

Cross-environment VPR benchmarks Method	Dim	Hawk	Laurel	17Pl	Gard	Baidu	N-Air	N-AirR	VP-Air	M-Atl	mean
Training-free											
CLS token	1024	26.3	50.9	61.1	75.0	43.1	56.3	76.1	38.0	24.8	50.2
GeM	1024	56.8	50.9	64.0	89.5	46.2	81.7	85.9	42.2	23.8	60.1
AnyLoc	49152	65.2	61.6	65.0	95.5	75.2	76.1	85.9	66.7	34.6	69.5
EffoVPR-ZS†	1024	38.1	67.0	**65.5**	97.5	75.9	83.1	88.7	65.7	36.6	68.7
SegVLAD-PreT	49152	56.8	26.8	64.8	91.5	78.5	45.1	40.9	69.8	22.8	55.2
Ours_{D2}†	1024	**67.0**	**83.0**	**65.5**	**99.5**	**84.5**	83.1	94.4	72.2	**37.6**	76.3
Ours_{D3}†	1024	61.0	75.9	65.3	**99.5**	83.5	**91.5**	**97.2**	**77.7**	**37.6**	**76.6**

Table 9: Full version of Table 3.

Standard VPR benchmarks Method	Dim	Pitts250k	Pitts30k	Tokyo24/7	MSLS-val	Nordland	St Lucia	Oxford	AmsterTime	mean
Trained or fine-tuned										
NetVLAD	32768	81.5	82.1	54.9	53.1	6.4	49.9	34.6	20.6	47.9
SFRS	4096	90.7	70.9	85.4	59.3	8.8	58.6	29.3	19.3	52.8
CosPlace	2048	92.2	90.9	87.3	87.7	63.0	99.9	79.1	49.7	81.2
MixVPR	4096	94.6	91.6	86.3	88.0	76.6	99.6	90.6	40.9	83.5
EigenPlaces	2048	94.1	92.5	93.0	89.1	63.6	99.5	89.5	48.9	83.8
R2Former†	256	93.2	91.1	88.6	89.7	60.5	99.7	92.7	29.5	80.6
SALAD	8448	95.1	92.4	94.6	92.2	86.5	**100.0**	97.9	58.4	89.6
CricaVPR‡	10752	94.6	91.8	90.2	89.2	80.5	99.8	98.4	49.0	86.7
SelaVPR†	1024	95.7	92.8	94.0	90.8	85.2	99.8	99.0	50.3	88.5
BoQ	12288	96.6	93.7	98.1	93.8	90.6	**100.0**	99.5	63.0	91.9
CliqueMining	8448	95.2	92.6	96.8	94.2	95.9	**100.0**	98.4	57.7	91.3
SuperVLAD‡	3072	95.2	92.6	93.7	91.5	79.2	99.9	**100.0**	54.7	88.3
VLAD-BuFF	4096	95.6	92.5	96.5	92.4	84.9	**100.0**	99.0	61.4	90.3
MegaLoc	8448	96.4	94.1	96.5	91.0	94.3	**100.0**	99.0	62.5	91.7
SAGE‡	8448	**96.8**	**94.7**	97.8	**94.5**	95.1	99.9	99.5	66.0	93.0
Ours_{D2} †	1024	95.7	92.5	97.1	89.0	82.3	99.0	99.0	68.2	90.3
Ours_{D3} †	1024	96.1	93.7	**98.4**	89.3	**96.0**	99.7	99.0	**73.6**	**93.2**

Cross-environment VPR benchmarks

Cross-environment VPR benchmarks Method	Dim	Hawk	Laurel	17Pl	Gard	Baidu	N-Air	N-AirR	VP-Air	M-Atl	mean
Trained or fine-tuned											
NetVLAD	32768	38.1	42.9	63.8	66.0	57.8	9.9	39.4	14.7	26.7	39.9
SFRS	4096	29.7	21.4	59.1	55.0	51.3	0.0	47.9	9.4	9.9	31.5
CosPlace	2048	33.9	25.9	63.3	84.5	47.1	0.0	76.1	10.4	32.7	41.5
MixVPR	4096	26.3	32.1	63.8	92.0	68.4	39.4	78.9	12.9	24.8	48.7
EigenPlaces	2048	28.8	27.7	63.8	86.0	66.6	12.7	**100.0**	13.1	29.7	47.6
R2Former†	256	17.8	30.4	63.8	84.0	74.7	26.8	83.1	15.0	22.8	46.5
SALAD	8448	38.1	46.4	64.3	96.5	69.9	53.5	94.4	24.0	15.8	55.9
CricaVPR‡	10752	28.0	34.8	64.0	94.0	65.8	45.1	**100.0**	19.9	27.7	53.3
SelaVPR†	1024	25.4	53.6	63.3	94.5	68.5	59.1	83.1	32.6	21.8	55.8
BoQ	12288	33.9	60.7	64.3	98.0	77.4	62.0	91.5	29.5	35.6	61.4
CliqueMining	8448	22.9	38.4	65.3	93.0	69.1	62.0	84.5	22.9	35.6	54.9
SuperVLAD‡	3072	37.3	51.8	64.8	95.5	70.3	14.1	88.7	23.8	13.9	51.1
VLAD-BuFF	4096	37.3	33.9	64.5	96.5	76.4	57.7	88.7	33.3	27.7	57.3
MegaLoc	8448	37.3	55.4	64.8	95.5	79.4	85.9	81.7	45.6	**40.6**	65.1
SAGE‡	8448	27.1	46.4	64.0	93.5	74.6	85.9	84.5	39.1	24.8	60.0
Ours_{D2} †	1024	**67.0**	**83.0**	**65.5**	**99.5**	**84.5**	83.1	94.4	72.2	37.6	76.3
Ours_{D3} †	1024	61.0	75.9	65.3	**99.5**	83.5	**91.5**	97.2	**77.7**	37.6	**76.6**

B.2 Visualization of the Same-Backbone Comparison

Figure 7 presents a visualisation of the results, listed in Table 6, when comparing the trained baselines with the proposed method in Section 4.4

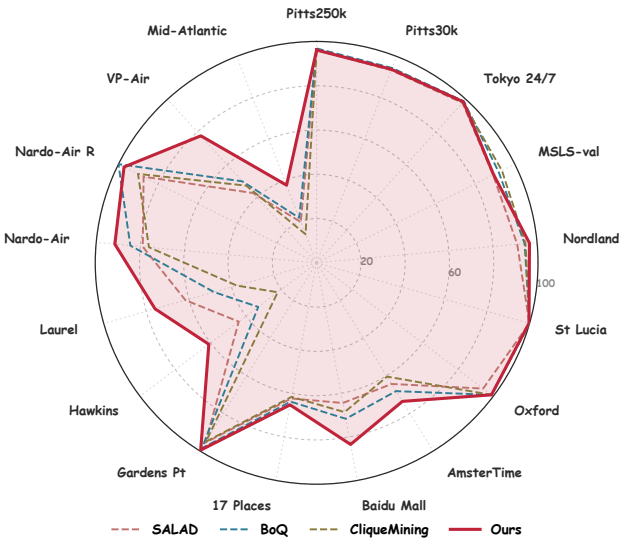

Figure 7: Table 6 as a radar chart.

B.3 FULL RESULTS: RECALL AT 1, 5 AND 10

In this section, we present the full version of results at Recall@5 and Recall@10 for Tables 3, 6 and 5.

Table 10: Recall@1, 5 and 10 of our pipeline on all 17 benchmarks. Ours_{D2} and Ours_{D3} use a frozen DINOv2 and DINOv3 backbone.

	Ours_{D2}			Ours_{D3}		
Benchmark	R@1	R@5	R@10	R@1	R@5	R@10
Standard VPR benchmarks						
Pitts250k	95.7	98.7	99.3	96.1	99.0	99.5
Pitts30k	92.5	96.9	97.9	93.7	97.3	98.1
Tokyo 24/7	97.1	99.7	100.0	98.4	99.7	100.0
MSLS-val	89.0	94.2	95.4	89.3	95.1	96.0
Nordland	82.3	89.9	92.1	96.0	98.2	98.7
St Lucia	99.0	99.7	99.8	99.7	99.9	99.9
Oxford	99.0	100.0	100.0	99.0	100.0	100.0
AmsterTime	68.2	87.4	91.1	73.6	92.3	94.6
Cross-environment VPR benchmarks						
Hawkins	67.0	94.1	98.3	61.0	92.4	99.2
Laurel	83.0	95.5	98.2	75.9	94.6	98.2
17 Places	65.5	82.0	90.6	65.3	81.3	89.9
Gardens	99.5	100.0	100.0	99.5	100.0	100.0
Baidu	84.5	93.2	96.0	83.5	93.5	96.3
Nardo-Air	83.1	98.6	100.0	91.5	100.0	100.0
Nardo-Air R	94.4	100.0	100.0	97.2	100.0	100.0
VP-Air	72.2	80.1	83.6	77.7	84.5	86.7
Mid-Atlantic	37.6	65.3	87.1	37.6	65.3	90.1

Table 11: Table 6 with Recall@1, 5 and 10.

	SALAD			BoQ			CliqueMining			Ours		
Benchmark	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Standard VPR benchmarks												
Pitts250k	96.2	99.0	99.5	96.8	99.3	99.7	96.1	99.0	99.6	96.1	99.0	99.5
Pitts30k	93.3	97.0	97.8	94.4	97.5	98.2	93.6	97.1	98.1	93.7	97.3	98.1
Tokyo 24/7	97.8	99.7	100.0	98.7	100.0	100.0	98.1	98.7	99.4	98.4	99.7	100.0
MSLS-val	89.0	96.1	97.4	91.9	96.8	97.4	93.5	97.2	98.2	89.3	95.1	96.0
Nordland	91.0	96.6	98.0	94.3	97.6	98.5	94.8	98.1	98.8	96.0	98.2	98.7
St Lucia	99.9	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	99.7	99.9	99.9
Oxford	94.2	99.5	100.0	99.0	100.0	100.0	99.0	100.0	100.0	99.0	100.0	100.0
AmsterTime	64.3	87.2	90.6	68.2	89.7	92.6	60.4	82.1	86.9	73.6	92.3	94.6
Cross-environment VPR benchmarks												
Hawkins	44.1	81.4	84.8	33.0	69.5	83.0	22.0	48.3	59.3	61.0	92.4	99.2
Laurel	61.6	87.5	92.0	48.2	78.6	87.5	37.5	72.3	86.6	75.9	94.6	98.2
17 Places	62.3	80.0	87.9	63.8	81.8	88.2	61.6	78.3	87.4	65.3	81.3	89.9
Gardens	95.5	99.5	100.0	98.5	99.5	100.0	95.0	99.5	100.0	99.5	100.0	100.0
Baidu	64.4	80.5	86.7	71.7	84.1	89.6	68.6	83.2	88.4	83.5	93.5	96.3
Nardo-Air	78.9	100.0	100.0	84.5	100.0	100.0	76.1	100.0	100.0	91.5	100.0	100.0
Nardo-Air R	87.3	100.0	100.0	100.0	100.0	100.0	90.1	100.0	100.0	97.2	100.0	100.0
VP-Air	42.6	59.6	67.5	49.8	67.3	75.0	47.2	66.1	74.0	77.7	84.5	86.7
Mid-Atlantic	19.8	57.4	69.3	21.8	45.5	54.5	13.9	47.5	56.4	37.6	65.3	90.1

Table 12: Full version of Table 5, per benchmark. Bold is the best in a row, underline the second best.

Benchmark	MaxSim	rapid sp.	cross-ref.	IDF	norm	attention	centrality	self-sim.
Pitts250k	89.2	90.5	72.9	83.2	90.3	95.7	94.0	95.7
Pitts30k	86.4	87.3	63.3	79.4	87.7	93.0	90.9	92.5
Tokyo 24/7	99.0	95.6	59.0	94.3	97.8	96.5	97.8	97.1
MSLS-val	60.4	66.4	26.5	41.1	71.2	80.7	81.8	89.0
Nordland	59.0	72.8	3.8	41.5	52.8	74.2	72.2	82.3
St Lucia	98.3	95.2	28.2	79.8	99.7	98.1	99.7	99.0
Oxford	100.0	99.5	30.9	100.0	99.5	96.3	100.0	99.0
AmsterTime	60.5	54.3	15.0	39.6	64.3	68.1	68.9	68.2
Hawkins	67.8	37.3	22.9	72.9	67.8	39.0	57.6	67.0
Laurel	75.0	46.4	46.4	80.4	75.9	64.3	80.4	83.0
17 Places	65.8	64.8	62.3	65.3	65.8	65.8	65.5	65.5
Gardens	99.0	98.5	85.0	99.0	99.0	98.0	99.0	99.5
Baidu	84.3	79.7	42.7	82.3	84.3	81.9	83.5	84.5
Nardo-Air	87.3	64.8	47.9	88.7	90.1	87.3	90.1	83.1
Nardo-Air R	94.4	95.8	67.6	83.1	95.8	95.8	95.8	94.4
VP-Air	61.2	41.2	22.7	59.3	61.2	71.8	69.3	72.2
Mid-Atlantic	33.7	39.6	4.0	33.7	34.7	41.6	37.6	37.6
mean (8)	81.6	82.7	37.5	69.9	82.9	87.8	<u>88.2</u>	**90.3**
mean (9)	74.3	63.1	44.6	73.8	75.0	71.7	<u>75.4</u>	**76.3**
mean (17)	77.7	72.3	41.2	72.0	78.7	79.3	<u>81.4</u>	**82.9**

Table 13: Recall@1, 5 and 10 on 3 benchmarks of Table 12.

	Pitts30k			AmsterTime			Nordland		
Patch importance signal	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
MaxSim	86.4	92.3	94.2	60.5	81.0	86.1	59.0	71.5	76.6
rapid spatial verification	87.3	93.5	95.4	54.4	72.5	76.3	72.8	84.7	88.5
cross-referencing	63.3	73.9	77.4	15.0	27.3	34.6	3.8	8.3	12.2
inverse document frequency	79.4	87.5	90.7	39.6	59.0	65.6	41.5	53.1	58.4
token norm	87.7	92.9	94.2	64.3	85.4	89.5	52.8	67.2	73.5
class-token attention	93.0	96.7	98.0	68.1	89.4	92.0	74.2	84.2	87.5
patch-graph centrality	90.9	95.0	96.4	68.9	88.5	91.8	72.2	82.6	86.4
self-similarity (ours)	92.5	96.8	97.8	68.2	87.5	91.3	82.3	89.9	92.1

C Qualitative Analysis

In this section, additional qualitative results are provided.

C.1 Additional qualitative results

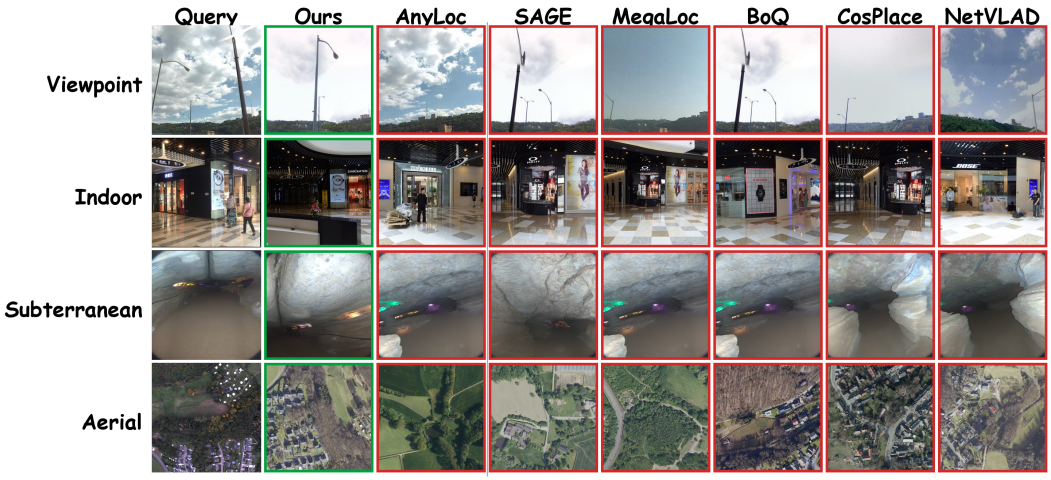

Figure 8: Rank-1 retrievals on 4 further benchmarks, one hard query each: Pitts30k, Baidu Mall, Laurel Caverns and VP-Air. Green marks a correct rank-1, red an incorrect one; training-free methods are left of the line.

C.2 Duplicated patches on every benchmark

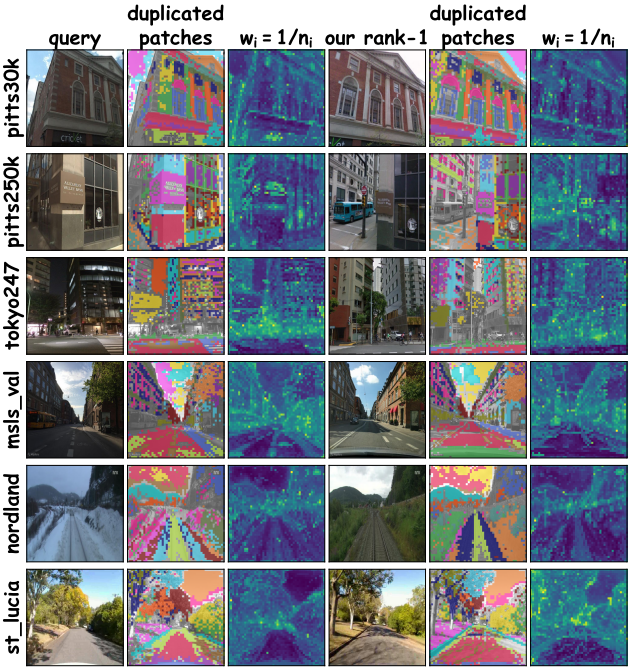

Figure 9: Full version of Figure 3, benchmarks 1–6.

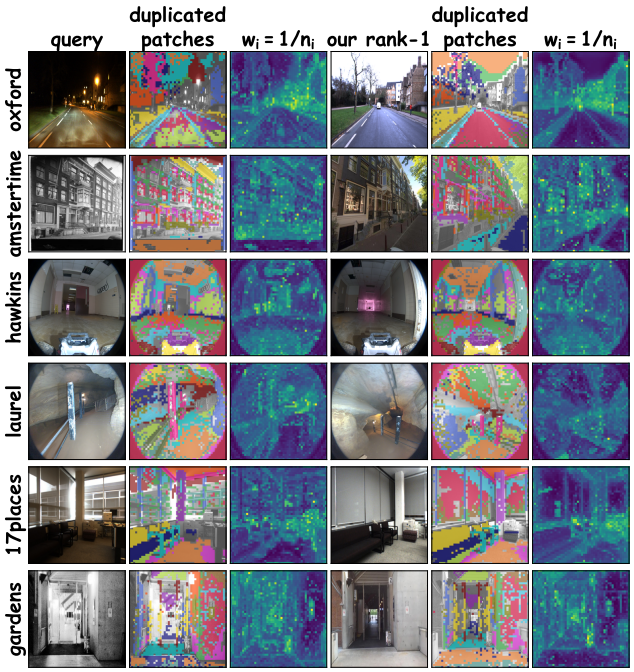

Figure 10: Full version of Figure 3, benchmarks 7–12.

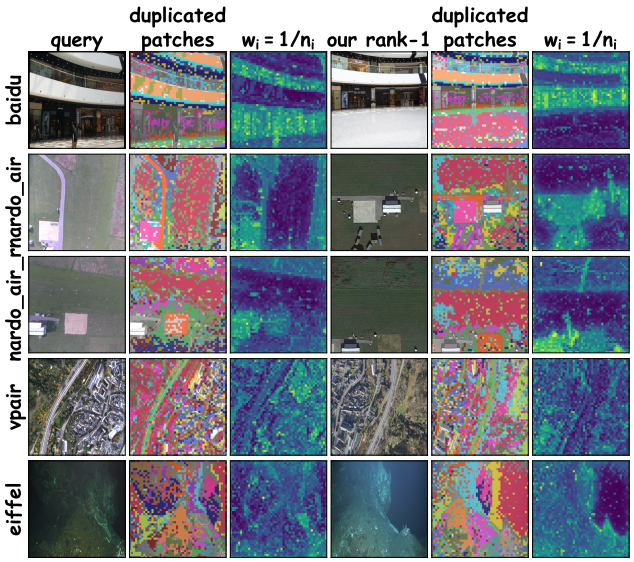

Figure 11: Full version of Figure 3, benchmarks 13–17.

D Limitation and Futurework

The total running time per query for UniPS is similar to that of many other baselines. However, we note that most of this running time is spent on feature extraction. To obtain robust patch features in a training-free pipeline, we rely on large vision backbones such as DINOv2 and DINOv3. This limits retrieval efficiency compared with methods that use smaller, trained backbones. As future work, we plan to explore data-free knowledge distillation to obtain a smaller backbone, accelerating the proposed training-free framework.