

Spatial Structure in UAV Visual Geolocalization: Controlled Evaluation of Descriptor Pooling and Geographic Index Partitioning

Ivan Suchý

`ivan.suchy@aos.sk`

Armed Forces Academy of general Milan Rastislav Štefánik Liptovský Mikuláš

Michal Turčaník

Armed Forces Academy of general Milan Rastislav Štefánik Liptovský Mikuláš

Article

Keywords: Visual geolocalization, UAV, GNSS-denied navigation, feature aggregation, vision foundation models, image retrieval, edge computing

Posted Date: October 19th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11232244/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Spatial Structure in UAV Visual Geolocalization: Controlled Evaluation of Descriptor Pooling and Geographic Index Partitioning

Ivan Suchý (Corresponding author)

Department of Informatics

Armed Forces Academy of general Milan Rastislav Štefánik

Liptovský Mikuláš, Slovakia

ivan.suchy@aos.sk

ORCID 0009-0000-1761-4795

Michal Turčaník (Corresponding author)

Department of Informatics

Armed Forces Academy of general Milan Rastislav Štefánik

Liptovský Mikuláš, Slovakia

michal.turcanik@aos.sk

ORCID 0000-0002-6773-213X

Abstract— Absolute visual localization lets a UAV estimate its position without GNSS by matching onboard imagery against a georeferenced map. Two design assumptions are widespread: that encoding where a feature appears improves the descriptor, and that partitioning the index geographically reduces retrieval cost. Direct evidence for either design choice remains limited because comparisons often lack an appropriately matched baseline: descriptor studies can confound spatial structure with patch-token use, while hierarchical indices are seldom evaluated directly against exhaustive search. We evaluate both assumptions using a frozen vision foundation backbone on synthetic queries and 1104 frames from three UAV flights, with end-to-end profiling on an embedded GPU. Neither assumption is supported under the evaluated configuration. Radially structured token pooling, at eight times the dimensionality of a plain global descriptor, is significantly worse on both query sets, and a patch-token mean-pooling control shows that the deficit is not explained by patch-token use alone. Aggregating the query over rotations increases inference time by 3.70 times without improving accuracy. Geographic partitioning increases median error by up to a factor of eight while occupying more memory and running slower on the target device. A 384-dimensional class token in a flat exact index localizes 120 m real-flight frames to a median of 68.7 m; on the edge device, the corresponding pipeline runs at 136 ms per frame, 1.65 J per frame and 12.1 W. Data and code are released.

Keywords—Visual geolocalization, UAV, GNSS-denied navigation, feature aggregation, vision foundation models, image retrieval, edge computing

I. INTRODUCTION

GNSS can be attenuated, jammed or spoofed; while inertial and visual odometry drift without bound. Therefore, a UAV requires a periodic absolute fix. Recovering that fix by matching onboard imagery against a georeferenced map is surveyed in [1]; the dominant formulation is retrieval followed by geometric refinement [2], [3]. Retrieval bounds everything downstream, since refinement cannot recover from a candidate drawn from the wrong region. Its known difficulty, distortion of the query relative to a nadir orthophoto, is addressed by viewpoint-robust dense features [4] and by rotation- and scale-invariant representations [5]. A parallel formulation uses supervised cross-view retrieval on paired benchmarks [6], [7], whose built-environment imagery and task-trained methods make absolute errors incomparable with the present untrained frozen-backbone setting over agricultural/peri-urban orthophoto terrain.

Frozen DINOv2 [8] and DINOv3 [9] features transfer without fine-tuning, and unsupervised aggregation over them matches task-specific place-recognition pipelines [10]; aerial geolocalization has adopted the pattern in two-phase systems [11] and in adapters that outperform full fine-tuning on limited data [12].

Given a frozen backbone, two design choices remain, namely how to pool patch features into one descriptor, and how to organize the index. Both are commonly resolved by adding spatial structure, in the descriptor through training-free re-weighting [13] or learned partitioning [14], [15], and in the index by searching only a geographically selected partition. Neither is routinely tested against a null hypothesis. Partition-based aggregators are compared against other aggregators rather than against a global descriptor of equal dimensionality, so a gain over a class token does not separate partitioning from patch-token use; and hierarchical index designs are reported within complete systems [2], [11] without measuring the cost of the hierarchy against exhaustive search on the same data. Both contrast with the direction of the aggregation literature, in which generalized-mean pooling [16], optimal-transport assignment [17], compact VLAD variants [18], learnable-query aggregation [19] and efficient decoders [20] pursue equal or better recall at reduced dimensionality.

This work evaluates both choices on an embedded deployment; Fig. 1 summarizes the pipeline. A frozen DINOv3 ViT-S/16 encodes an 18.29 km² orthophoto of northern Slovakia, discretized into 3213 overlapping tiles, and each query frame. Four descriptor configurations are compared under identical conditions: the class token (CLS) and global mean pooling (mean-patch) at 384 dimensions, radial token pooling at 3072 dimensions, and a query-side multi-rotation variant re-using the radial map descriptor. Each is evaluated under exhaustive and partitioned retrieval, on 200 synthetic queries and 1104 frames from three flights, and profiled on an NVIDIA Jetson Orin Nano. Under the evaluated backbone, tiling scheme, map size, retrieval implementation, and edge platform, neither spatially structured pooling nor geographic index partitioning provides a net accuracy or edge-efficiency benefit over flat exhaustive retrieval.

Our contributions are the following.

- (1) A descriptor ablation with a patch-token control for the evaluated radial construction.
- (2) A measurement of the cost of hierarchical partitioning under both backends on identical data, with a coverage-matched sweep, showing that most of the apparent descriptor degradation is due to the retrieval backend.
- (3) Full-pipeline edge profiling under pinned clocks and thermal steady state, including the measured effect of half-precision inference.
- (4) A quantified account of the synthetic-to-real gap, derived from the camera model in advance of the results.
- (5) An original released dataset of UAV

video with telemetry, with the reference map and the core pipeline.

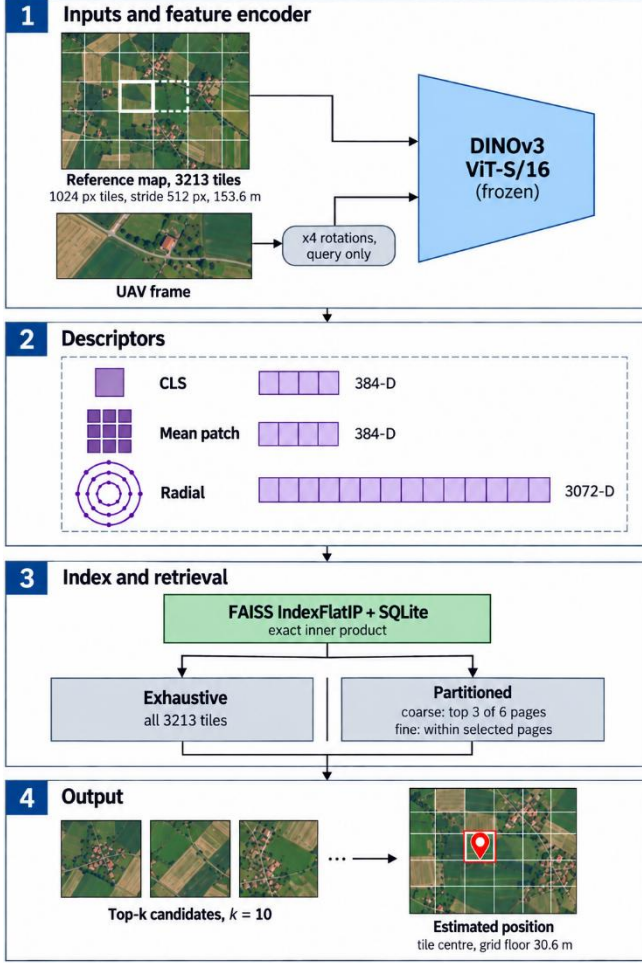


Fig. 1. Overview of the evaluated pipeline, from reference map and query frame to position estimate.

II. RELATED WORK

Absolute visual localization for UAVs is surveyed in [1]. Methods divide into template matching, sparse feature matching with geometric verification, and retrieval of a georeferenced patch using a global descriptor; contemporary systems combine the last two coarse-to-fine [2], [3], [21]. A recurring concern is geometric mismatch, addressed by orthorectification against a digital elevation model [22], rotation-equivariant dense features [4], terrain-weighted optimization [23], and vector rather than raster references [5]. The recurrence of these mechanisms across independent studies suggests that acquisition geometry, rather than descriptor capacity, is the dominant deployment constraint, a conclusion Section VI-C reaches from a different direction. The closest prior system is [2], which pairs a frozen backbone and generalized-mean pooling with keypoint registration, inertial priors and a sliding-window map update; the present work shares its retrieval stage and frozen-backbone premise and stops there, since a refinement stage would compose with these results rather than alter their ordering.

A large literature sees UAV localization as cross-view retrieval between drone and satellite imagery, driven by paired benchmarks [6], [7], with contributions such as hard-negative sampling [24], self-supervised feature enhancement [25], shifting-dense partition learning [14], direction-guided multiscale fusion [15], semantic graph representations [26],

training without ground truth [27] and the use of flight status such as height [28]. These task-trained methods are not directly comparable with the present untrained frozen-backbone setting over agricultural and peri-urban orthophoto terrain. Accordingly, only the relative comparison between aggregation schemes transfers. Partition-based aggregation [14], [15] is the closest family to the radial descriptor, but both are task-trained, unlike the fixed untrained radial partition used here. The negative result of Section V is therefore specific to this training-free construction. Altitude dependence is treated in multi-height benchmarks [7], by partitioning robust to target shift and scale [14], and by feeding flight height to the network as an auxiliary input [28]; these bear on Sections IV-B and VI-C.

Large self-supervised vision transformers have become the standard frozen feature extractor. DINOv2 [8] transfers to dense prediction untouched, AnyLoc [10] showed that such features with unsupervised aggregation match task-specific pipelines, and DINOv3 [9], used here in a ViT-S/16 variant, provides a compact backbone for evaluating embedded deployment. The hierarchical system in [2] uses a frozen retrieval backbone; AirGeoNet [11] provides a related map-guided aerial localization formulation. Given the available data and the scope of this study, the backbone is kept frozen. Under AdaptGeo’s cross-view protocol [12], full DINOv2 fine-tuning reduces top-1 retrieval from 88.1 % to 21.6 %, whereas a lightweight adapter exceeds the frozen baseline, motivating parameter-efficient adaptation here. These models carry register tokens [29], which change the output sequence layout, so treating registers as patch tokens would contaminate any pooled descriptor.

How patch features are pooled is the central design question. Generalized-mean pooling [16] and VLAD-style aggregation remain the standard unsupervised baselines, paired with frozen features in [10] and [11]. Spatial structure can be imposed without training. GeoCluster [13] re-weights a NetVLAD assignment map with a centre-favouring mask and improves recall on aerial datasets. It is the closest analogue to the radial descriptor, but its central prior contrasts with equal-dimensional annuli of unequal area, which allocate substantial capacity to peripheral regions. Section VI-A returns to this distinction. Learned aggregators improved on both baselines through optimal-transport assignment [17], compact VLAD variants [18], learnable queries [19], cross-image correlation [30] and efficient decoders [20]. Recent work has increasingly emphasized compact descriptors, because retrieval cost and storage scale linearly with descriptor size, and the 3072-dimensional radial descriptor runs against the trend. Where a fixed prior is imposed on a frozen backbone there is no training signal to discover that it is useless, so detection falls to ablation.

Retrieval itself is a nearest-neighbour problem, with exact and quantized indices in wide use [31]; partitioning geographically rather than in descriptor space is motivated by trajectory continuity and appears in [2] and [11]. Onboard deployment imposes power, unified-memory, and thermal constraints that reducing descriptor size can partly mitigate [18], [20]. Quantitative onboard profiling of a frozen backbone driving georeferenced retrieval is nonetheless sparsely reported, since published edge figures are typically inference latencies for a converted engine measured in isolation. Section V-D instead measures the complete query path under pinned clocks and thermal steady state.

III. METHODOLOGY

This section defines the system, from the problem statement and error metric (Section III-A) to the implementation (Section III-F). No component is trained.

A. Problem formulation

Let $\mathcal{M}$ be a georeferenced orthophoto mosaic on a bounded region $\Omega \subset \mathbb{R}^2$ in EPSG:5514 (S-JTSK / Krovák East North), and a UAV observation an RGB image I from a nadir-facing camera at unknown planimetric position. The map is discretized into tiles $\mathcal{T} = \{t_1, \dots, t_N\}$ with centres $\mathbf{c}_i \in \Omega$, and a descriptor function ϕ maps tiles and queries onto the unit hypersphere $\mathbb{S}^{d_\phi-1}$, where d_ϕ is the strategy-dependent descriptor dimensionality. The estimated position is the centre of the tile maximizing cosine similarity,

$$\hat{\mathbf{c}}(I) = \mathbf{c}_{i^*}, \quad i^* = \arg\max_{i \in \mathcal{C}(I)} \langle \phi(I), \phi(t_i) \rangle \quad (1)$$

where $\mathcal{C}(I) \subseteq \{1, \dots, N\}$ is the candidate set produced by the retrieval backend; $\mathcal{C}(I) = \{1, \dots, N\}$ recovers exhaustive search. The localization error against ground truth $\mathbf{g}$ is $e(I) = \|\hat{\mathbf{c}}(I) - \mathbf{g}\|_2$, reported in metres. No sub-tile refinement or top- k interpolation is applied, so (1) characterizes the retrieval stage in isolation.

B. Reference map and tile grid

The reference map is a subset of Ortofotozozka SR, Stred 2024 (RGB) [32] covering the Liptovský Mikuláš region at a ground sampling distance of $g = 0.15$ m/px. All geometry is expressed in EPSG:5514. Because the local scale factor is close to unity over the study area, tile, stride, and page dimensions can be treated as metric distances. Four map sheets are mosaicked into a single three-band GeoTIFF; the near-infrared band is discarded.

Let (X_L, Y_T) be the raster origin, its upper-left corner, with rows advancing in the direction of decreasing y . Tile centres lie on a lattice of spacing s given by (2),

$$\mathbf{c}_{(u,v)} = (X_L + \delta_x + u s, \quad Y_T - \delta_y - v s) \quad (2)$$

for $u \in \{0, \dots, N_x - 1\}$ and $v \in \{0, \dots, N_y - 1\}$, with $\delta_x = 153.6$ m and $\delta_y = 76.8$ m. The lattice is a QGIS point layer exported to a catalogue builder, which resolves each centre to a $\tau_{px} \times \tau_{px}$ raster window; points whose window would extend beyond the raster are rejected, so N counts only tiles with a complete footprint. The working raster has dimensions of $33\,334 \times 26\,668$ px, contains three bands, and is represented in EPSG:5514, spanning 5000.1×4000.2 m. Tiles are $\tau_{px} = 1024$ px ($\tau = 153.6$ m) at stride $s_{px} = 512$ px ($s = 76.8$ m) on a 63×51 grid, giving $N = 3213$ tiles whose centres span $A = 4761.6 \times 3840.0$ m (18.29 km²), to which all density figures refer; the network input is 512×512 px. With $\tau = 2s$ the footprints overlap by 50 % in each axis.

Because (1) returns a tile centre, the lattice bounds the median error attainable by a retriever that always returns the nearest centre. For a query position uniform over a lattice cell, $\Pr[R \leq r] = \pi r^2 / s^2$ for $r \leq s/2$, giving (3).

$$\text{med}(R) = \frac{s}{\sqrt{2\pi}} = 30.6 \text{ m}, \quad \text{max}(R) = \frac{s}{\sqrt{2}} = 54.3 \text{ m} \quad (3)$$

Equation (3) provides the quantization reference for interpreting the reported medians; success rates near 30 m are

strongly constrained by the tile-centre discretization and should therefore be interpreted with caution.

C. Spatial partition

The map is partitioned into square pages, each owning a disjoint subset of tiles and its own index, on a lattice anchored at the raster origin and defined geometrically by (4),

$$\pi_P(x, y) = \left(\left\lfloor \frac{x - X_L}{P} \right\rfloor, \left\lfloor \frac{Y_T - y}{P} \right\rfloor \right) \quad (4)$$

the y term reversed because rows advance downward. A tile is assigned to the page containing its centre, so $\mathcal{T} = \sqcup_p \mathcal{P}_p$ with $\mathcal{P}_p = \{t_i: \pi_P(\mathbf{c}_i) = p\}$: each tile belongs to exactly one page even though footprints overlap and may cross page boundaries. With $L_x^c = 4915.2$ m and $L_y^c = 3916.8$ m the maximum tile-centre offsets, the number of non-empty pages is $(\lfloor L_x^c/P \rfloor + 1)(\lfloor L_y^c/P \rfloor + 1)$. Three configurations are evaluated, all over $N = 3213$ tiles: $P = 2048$ m giving 6 pages on a 3×2 grid with $275 / 637.5 / 702$ tiles per page (min / median / max), $P = 1000$ m giving 20 pages on 5×4 with $144 / 156.0 / 169$, and $P = 600$ m giving 63 pages on 9×7 with $10 / 56.0 / 64$. All pages are non-empty. Page assignment is a property of the catalogue, not of the descriptors, so changing P rebuilds only the catalogue and the index.

The coarse stage selects k_p pages, making a fraction of the index reachable,

$$\rho = \frac{k_p}{n_{\text{pages}}(P)} \quad (5)$$

The number of vectors scored in the fine stage is approximately ρN , exactly so only under uniform page occupancy, which does not hold here. At fixed $k_p = 3$ the three configurations give $\rho = 0.500, 0.150$ and 0.048 , so varying P at fixed k_p varies granularity and search budget simultaneously; Section V-C therefore reports a sweep at matched ρ of (5).

D. Descriptors

All descriptors derive from a single frozen DINOv3 ViT-S/16 (facebook/dinov3-vits16-pretrain-lvd1689m) [9], with identical weights for map tiles and queries, following established practice [2], [10], [11]. No fine-tuning or adaptation is performed.

Images are resized to 512×512 px with bilinear interpolation and normalized with the checkpoint's configuration, which applies no centre crop and uses ImageNet statistics. With patch size 16 this gives $G = 32 \times 32 = 1024$ patch tokens of dimension $D = 384$, preceded by a class token and $n_r = 4$ register tokens [29]; patch tokens are addressed from index 5, and registers carry no grid position and must not enter any pooling. Map tiles are read as 1024 px windows at g and resized to 512 px, so the database and the synthetic query set operate at an effective 0.30 m/px; real frames do not (Section IV-B).

Write $\mathcal{N}(\mathbf{v}) = \mathbf{v} / \max(\|\mathbf{v}\|_2, \epsilon)$ with $\epsilon = 10^{-12}$. Two 384-dimensional baselines are evaluated: The first is the normalized class-token descriptor (CLS), $\phi_{\text{CLS}} = \mathcal{N}(\mathbf{h}^{\text{cls}})$. The second is global mean pooling over patch tokens (mean-patch), $\phi_{\text{mean}} = \mathcal{N}(1/G \sum_j \mathcal{N}(\mathbf{h}_j^{\text{pat}}))$, where the inner normalization averaging unit directions so that high-norm tokens cannot dominate.

The radial descriptor partitions the patch grid into R concentric annuli. With (r_j, c_j) the grid coordinates of token j and $\bar{r} = \bar{c} = (32 - 1)/2$ the centre, (6) gives the radius and bin index,

$$\begin{aligned} \varrho_j &= \sqrt{(r_j - \bar{r})^2 + (c_j - \bar{c})^2}, \\ \beta(j) &= \min(\lceil \varrho_j / \varrho_{\max} R \rceil, R - 1) \end{aligned} \quad (6)$$

with $\varrho_{\max} = \max_j \varrho_j$. Each bin is pooled by an arithmetic mean of normalized tokens and the per-bin vectors are concatenated and renormalized, as in (7),

$$\begin{aligned} \mathbf{z}_b &= 1/|\mathcal{B}_b| \sum_{j \in \mathcal{B}_b} \mathcal{N}(\mathbf{h}_j^{\text{pat}}), \\ \phi_{\text{rad}} &= \mathcal{N}([\mathbf{z}_0^\top, \dots, \mathbf{z}_{R-1}^\top]^\top) \in \mathbb{R}^{RD} \end{aligned} \quad (7)$$

where $\mathcal{B}_b = \{j: \beta(j) = b\}$; an empty bin yields $\mathbf{z}_b = \mathbf{0}$, a case that does not arise at $R = 8$ on a 32×32 grid. With $R = 8$ and $D = 384$ the descriptor has 3072 dimensions, eight times the baselines. Pooling within a bin is a mean rather than a generalized-mean aggregation [16], because DINOv3 patch tokens contain negative components and the standard formulation requires a clamp that would discard them.

Two properties of (6) are used in Section VI-A. The bins are equal in radius, not in area or token count, so tokens are distributed unevenly while each bin contributes the same D dimensions; and $\varrho_{\max}$ is the corner distance $\sqrt{2} \bar{r} \approx 21.92$, not the inscribed radius $\bar{r} = 15.5$, so bins 6 and 7 lie entirely outside the inscribed circle and contribute 768 of the 3072 dimensions (Fig. 2).

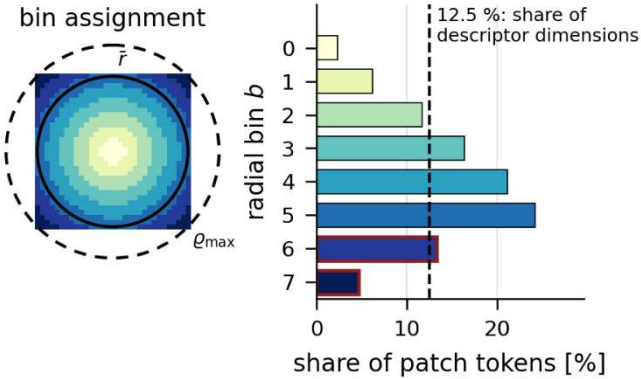


Fig. 2. Radial binning of the 32×32 patch grid at $R=8$. Left: bin assignment by (6), with the inscribed circle and the corner distance $\varrho_{\max}$. Right: share of tokens per bin against the constant 12.5% share of descriptor dimensions.

Rotating the token map by a multiple of 90° permutes tokens within bins and leaves (7) unchanged, so the pooling operator is invariant to such rotations; at intermediate angles tokens change radius on the discrete grid and may cross bin boundaries. The multi-rotation strategy applies the radial descriptor to rotated query tensors at query time, on the assumption that the composition with the encoder approximately inherits the property. The preprocessed query tensor is rotated by each $\alpha \in \{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$ with an exact tensor rotation requiring no interpolation; the database is not rotated and the index does not grow. Each rotated query is an independent two-stage search with its own candidate set $\mathcal{C}_\alpha(I)$, and scores are merged over the union by the maximum per tile, as in (8).

$$i^* = \arg \max_{i \in \bigcup \mathcal{C}_\alpha(I)} \max_{\alpha: i \in \mathcal{C}_\alpha(I)} \langle \phi_{\text{rad}}(\text{rot}_\alpha(I)), \phi_{\text{rad}}(t_i) \rangle \quad (8)$$

Inference cost therefore scales with the four rotations, and each rotation selects its own k_p pages, so the effective candidate set is up to four times larger than for single-rotation radial.

E. Index and retrieval

All vector indices are exact inner-product indices (IndexFlatIP in FAISS [31]), with vectors L2-normalized before insertion, so the index computes the cosine similarity of (1) and introduces no approximation error, preventing descriptor quality and quantization loss from being confounded. Three levels are built per strategy: a flat index over all N tile descriptors; one per-page index over $\phi(\mathcal{P}_p)$; and a coarse index over the page descriptors of (9),

$$\psi_p = \mathcal{N} \left(1/|\mathcal{P}_p| \sum_{t_i \in \mathcal{P}_p} \phi(t_i) \right) \quad (9)$$

Given a query descriptor $\mathbf{q}$, the k_p highest-scoring pages under the coarse index are selected, each is searched exhaustively for its own top k tiles, and the union is sorted by descending score with a stable sort, deduplicated by tile identifier keeping the first occurrence, and truncated to k , so equal scores resolve by coarse page rank then within-page index order and retrieval is deterministic. Throughout, $k_p = 3$ and $k = 10$.

Retrieval is exact within the selected pages and approximate across the map, since any difference from exhaustive search arises when the coarse stage excludes the page holding the exhaustive top-1 candidate. The coarse recall of a query set is the fraction of queries for which that page is selected, which for top-1 equals the fraction on which partitioned retrieval reproduces the exhaustive result. It is a fidelity measure defined per strategy against that strategy's own exhaustive baseline, not a cross-strategy measure of accuracy.

Per-page indices are loaded on demand into a least-recently-used cache of capacity m pages, governed by $\kappa = n_{\text{pages}}(P)/m$; for $\kappa \leq 1$ the whole index becomes resident after a transient. Optional neighbour prefetching is disabled in all accuracy experiments.

F. Implementation and platforms

Map preparation uses QGIS 3.40.13 with GDAL 3.12.0 and PROJ 9.7.0; raster windowing uses rasterio; descriptors use the Hugging Face transformers implementation; search uses FAISS. Database descriptors for all three variants over 3213 tiles were extracted on the workstation in float32 from native 1024×1024 px windows; index construction is an offline one-time cost, excluded from all latency figures. The workstation is an Intel x86-64 machine, 12 cores, 63.8 GiB RAM, PyTorch 2.12.0 CPU, faiss-cpu 1.14.3; the edge device is an NVIDIA Jetson Orin Nano Super (Tegra234, six Cortex-A78AE cores at 1.728 GHz, 7.4 GiB unified memory), L4T R36.5.2, CUDA 12.6, PyTorch 2.11.0, faiss-cpu 1.15.0.

Vector search runs on the CPU on both platforms; the GPU is used only for backbone inference on the edge device. Edge inference uses float16 while retrieval remains a float32 inner product, since descriptors are cast back before indexing; the effect is quantified in Section V-D. For edge measurements the device is set to MAXN_SUPER with clocks pinned at 1728, 1020 and 3199 MHz for CPU, GPU and memory, and

reaches thermal steady state before the matrix starts; each condition is repeated three times with per-stage timings, utilization, temperature and energy recorded by tegrastats at 100 ms.

IV. EXPERIMENTAL SETUP

Four descriptor configurations are compared under identical retrieval conditions: CLS and mean-patch at 384 dimensions, radial at 3072, and the multi-rotation variant. Including mean-patch is essential, since comparing a 3072-dimensional descriptor only against the class token confounds the pooling scheme with patch-token use, whereas global mean pooling over the same tokens tests whether this alone explains the radial deficit; unsupervised pooling over frozen features is also the standard baseline here [10], [11]. The database is identical throughout, $N = 3213$ tiles over 18.29 km^2 , built once per strategy, and nothing is trained.

A. Query sets

Two hundred synthetic queries are cut from the orthomosaic that forms the database, with controlled geometric and photometric perturbation, and provide a matched-domain reference for the evaluated pipeline. Centres are drawn uniformly from the AOI shrunk by a 200 m margin, keeping the oversampled read window of a rotated query inside the raster, and are unconditioned on the tile lattice, so (3) applies in full. Each query reads an oversampled ground window of 245.76 m at 819 px , applies rotation, scale σ and a brightness gain, and only then centre-crops to 512 px , so no out-of-frame region can enter. Ground coverage is given by (10); ground truth is the sampled centre, exact by construction.

$$\tau_q(\sigma) = \tau/\sigma \in [146.3, 161.7] \text{ m} \quad (10)$$

Three flights were flown with a DJI Mavic Pro, whose 28 mm equivalent lens records 1920×1080 video at 29.97 fps. Frames are decoded deterministically at 1 Hz and matched to the nearest DJI SRT telemetry block, subject to $|\Delta t| \leq 0.60 \text{ s}$ with no manual offset; the achieved median offset is 8 ms and the maximum 16 ms, under 0.2 m at 10 m/s. Matched positions are transformed from WGS-84 to EPSG:5514 and used as ground truth; no RTK or PPK correction was applied, so the contribution of GNSS error to the total error budget is not quantified.

Frames are excluded, by rules fixed before evaluation, if the telemetry match is invalid, the position falls outside the map, GNSS coordinates are missing, or the implied ground speed exceeds 40 m/s; these rules excluded no frames. A rule requiring the gimbal within 15° of nadir is inactive, because the telemetry carries no gimbal attitude field. The telemetry reports absolute altitude only; the nominal altitudes above ground level (AGL) are supported, since subtracting them from each flight's median absolute altitude yields 613.6, 612.3 and 614.2 m, agreeing to within 1.9 m. The flights `flight_080m_01`, `flight_100m_01` and `flight_120m_01` at nominal 80, 100 and 120 m AGL contribute 359, 380 and 365 frames over 3265, 3998 and 3561 m of track at median ground speeds of 9.5, 11.2 and 10.0 m/s, each traversing roughly 0.4 km^2 , about 2 % of the mapped area. Synthetic perturbations comprise rotation $U(\pm 30^\circ)$, scale $\sigma \sim U(0.95, 1.05)$ and brightness $U(0.85, 1.15)$. All perturbations use seed 42 and bilinear resampling.

B. Acquisition gap between the query sets

The two query sets are not samples of the same distribution, and Section V compares them only with the following differences stated, each quantified in advance from the acquisition geometry. A 28 mm-equivalent lens on a 16:9 frame subtends 65.5° horizontally and 39.8° vertically, so the ground footprint at altitude h is (11).

$$W(h) = 1.2857 h, \quad H(h) = 0.7232 h \quad (11)$$

A real query therefore covers 25 % to 57 % of the ground area of the map tile it must match, against 90–110 % for a synthetic query. Both tiles and queries are resized to a square network input, but map tiles are square to begin with, so a 16:9 frame is compressed along one axis by $16/9 = 1.778$ independently of altitude. Only the horizontal scale of the 120 m flight, 0.301 m/px , coincides with the database scale of 0.300 m/px .

A third difference, in-plane orientation, cannot be quantified because the telemetry contains no yaw field. Database tiles are map-north-up, but the telemetry carries no yaw field, so real queries are subject to uncontrolled in-plane rotation over $[0^\circ, 360^\circ)$ whereas synthetic queries are rotated within $\pm 30^\circ$. Image rotation is a known difficulty for map-based matching, which methods that obtain rotation and scale invariance by construction avoid [5]. Real frames additionally carry lossy compression, motion blur and illumination and seasonal conditions differing from the mosaic. Consequently only the ordering of descriptor strategies, not absolute error magnitude, is compared across the two sets.

C. Configurations, metrics and protocol

Unless stated otherwise every evaluation uses $k_p = 3$, $k = 10$, the $P = 2048 \text{ m}$ configuration, and a success tolerance of 30 m. Cache capacity is $m = 12$ on the workstation and $m = 5$ on the edge device; it governs page loading only and cannot alter which tiles are retrieved, since every resident page is searched exactly. Neighbour prefetching is inactive, since its trigger requires a query position that the accuracy pipeline never supplies. Three sweeps are performed: page size $P \in \{2048, 1000, 600\} \text{ m}$ at fixed k_p and at matched ρ ; cache capacity spanning κ from below to above unity; and exhaustive against partitioned retrieval on the same queries and descriptors.

The primary metric is the median of $e(I)$, preferred to the mean because the error distribution is heavy-tailed. Supporting metrics are the success-at-tolerance curve $\Pr[e(I) \leq \delta]$, reported as a curve for the reason given after (3); the catastrophic-failure rate $\Pr[e(I) > 500 \text{ m}]$; and coarse recall, which attributes partitioned-retrieval error to the coarse stage rather than the descriptor.

All descriptor comparisons are paired on the identical query set in the identical order. Synthetic comparisons use the Wilcoxon signed-rank test under Holm correction across the six comparisons, with top-1 agreement tested by McNemar and confidence intervals obtained by non-parametric bootstrap. Real-flight frames sampled at 1 Hz are strongly autocorrelated, so inferential uncertainty is assessed by temporal block bootstrap; frame-level Wilcoxon and McNemar results are retained as sensitivity analyses. The 1104 frames are partitioned into 56 contiguous 20 s blocks with 5000 block-level resamples; at 10 m/s a block spans roughly 200 m, more than one tile footprint.

Fifteen edge conditions, combining query set, strategy and backend, were each executed three times, for 45 runs with no failures; reported timings are means with the between-run standard deviation alongside. Edge runs use an instrumented variant of the evaluation scripts differing only in measurement, in that decoding is brought inside the timed region and the cache emits a load event on every demand load. Every aggregate in Section V is computed from per-query records. The released material is listed in the Data Availability Statement.

V. RESULTS

Results are presented in four parts: the descriptor comparison on both query sets (Section V-A), the same comparison under exhaustive search, which isolates the retrieval backend (Section V-B), partition granularity and cache capacity (Section V-C), and the edge profiling (Section V-D).

A. Descriptor comparison

Table I reports the four configurations on both query sets under the reference configuration, and Fig. 3 gives the corresponding success-at-tolerance curves.

TABLE I

DESCRIPTOR COMPARISON UNDER PARTITIONED RETRIEVAL ($P = 2048$ m, $k_p = 3$). SYNTHETIC: $n = 200$. REAL: $n = 1104$ FRAMES FROM THREE FLIGHTS

Strategy	Dim	Synth. med. [m]	Synth. $e > 500$ m	Real med. [m]	Real ≤ 30 m
mean-patch	384	41.78	0.220	1271.2	0.053
CLS	384	42.26	0.180	566.2	0.092
radial	3072	51.22	0.265	1756.8	0.040
multi-rotation	3072	52.52	0.265	1691.4	0.056

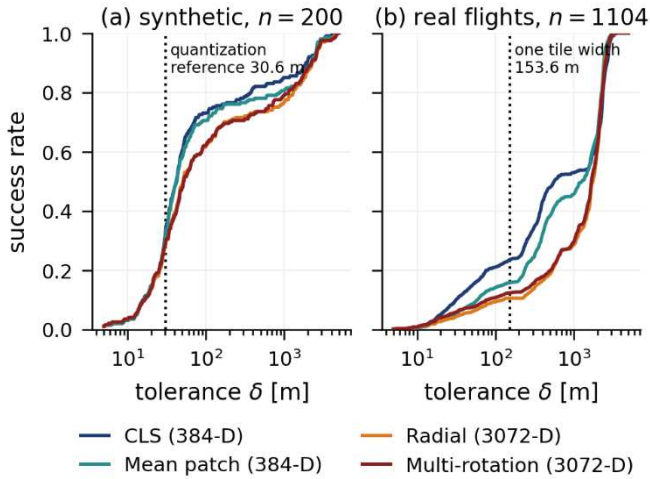


Fig. 3. Success rate against tolerance δ . (a) Synthetic queries, with the 30.6 m quantization reference of (3) marked. (b) Real flights, with one tile width marked.

The consistent ordering of the four strategies is the central empirical result. The two 384-dimensional global descriptors outperform the 3072-dimensional radial descriptor on every reported statistic on both query sets. Radial pooling costs a factor of eight in dimensionality and returns a synthetic median error 21 % higher, a catastrophic-failure rate 47 % higher in relative terms, and lower success rates across the reported operating tolerances. Fig. 3(a) also shows why the

comparison must be read as a curve, since below roughly 40 m the strategies are indistinguishable, the criterion near the reference of (3) being dominated by grid geometry.

Table II gives the paired tests. On the synthetic set every comparison between a 3072-dimensional variant and a 384-dimensional baseline is significant after Holm correction, while neither pair within a dimensionality class is. The disadvantage of radial pooling is therefore not an artefact of comparing patch-token pooling against the class token, since mean-patch uses the same tokens at the same dimensionality and beats radial at $p = 1.2 \times 10^{-3}$; the deficit is specific to the evaluated radial construction rather than to patch-token use. Multi-rotation provides no measurable benefit over its base despite evaluating the backbone four times and searching a candidate set up to four times larger.

TABLE II

PAIRWISE COMPARISONS. WILCOXON P-VALUES ARE HOLM-CORRECTED ACROSS SIX COMPARISONS; MCNEMAR USES THE 30 M CRITERION WITH DISCORDANT COUNTS A-ONLY / B-ONLY. REAL-FLIGHT FRAME-LEVEL TESTS ARE SENSITIVITY ANALYSES; INFERRENTIAL UNCERTAINTY IS ASSESSED BY TEMPORAL BLOCK BOOTSTRAP

Comparison	Synth. p	Real p	McNemar	Disc.
CLS vs mean-patch	0.271	0.287	1.9×10^{-9}	49 / 5
CLS vs radial	7.6×10^{-4}	1.2×10^{-17}	2.8×10^{-14}	61 / 3
CLS vs multi-rotation	0.0158	1.4×10^{-15}	1.8×10^{-6}	52 / 12
mean-patch vs radial	1.2×10^{-3}	5.5×10^{-19}	0.0187	20 / 6
mean-patch vs multi-rotation	0.0385	1.4×10^{-15}	0.636	18 / 22
radial vs multi-rotation	0.271	0.247	1.6×10^{-3}	4 / 22

On real flights the ordering is preserved and amplified, the gap between the two 384-dimensional baselines, indistinguishable on synthetic queries, opens to a factor of 2.2. The paired median difference is exactly 0 m for all six comparisons on both sets because many queries yield identical top-1 tiles across strategies, tying the per-query errors. On the 59 % of synthetic queries where all four strategies return the same tile their medians lie within 1.5 m, and the entire descriptor effect is carried by the discordant remainder, where the median error of radial is 4.8 times that of CLS. Correspondingly, CLS succeeds within 30 m on 49 real frames where mean-patch fails against 5 in reverse. The temporal block bootstrap preserves the direction and significance pattern of Table II, with wider uncertainty reflecting 56 effectively independent blocks.

Two secondary analyses yielded no statistically conclusive result. The rotation stratification orders the point estimates as the descriptor's construction predicts, radial degrading $1.46 \times$ between low and high bins against $1.14 \times$ for CLS, but every interval contains unity and the study is underpowered at $\pm 30^\circ$ and $n = 200$; and radial versus multi-rotation on real flights is not significant by Wilcoxon yet significant by McNemar on 26 of 1104 frames, which Section VI-A interprets.

B. Exhaustive against partitioned retrieval

Table III compares exhaustive against partitioned retrieval on identical queries and descriptors. On synthetic queries the ratio of partitioned to exhaustive median error is 1.30, 1.28, 1.44 and 1.45 for CLS, mean-patch, radial and multi-rotation, whose exhaustive medians are 32.60, 32.75, 35.65 and 36.28 m at coarse recall 0.735–0.760. Fig. 4 shows the comparison and its effect on the tail.

TABLE III

EXHAUSTIVE AGAINST PARTITIONED RETRIEVAL: MEDIAN ERROR [M] AND COARSE RECALL

Flight	Strategy	Exhaustive	Partitioned	Ratio	Recall
080m	CLS	250.0	461.1	1.84	0.688
080m	radial	614.2	1680.1	2.73	0.655
100m	CLS	127.5	615.7	4.83	0.632
100m	radial	939.4	1664.0	1.77	0.711
120m	CLS	68.7	575.9	8.38	0.573
120m	radial	1075.5	1948.4	1.81	0.627

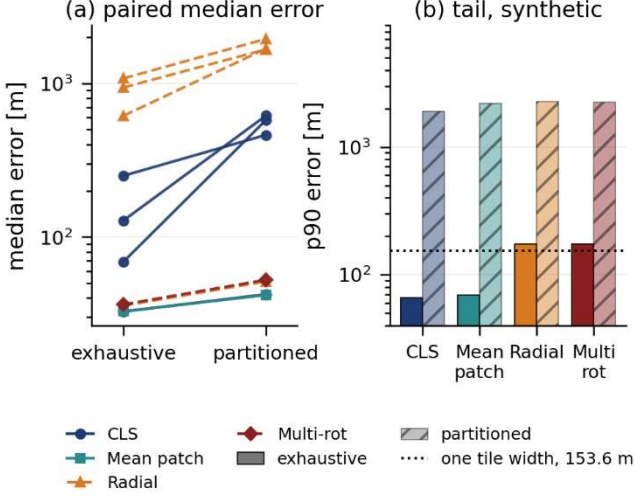


Fig. 4. Exhaustive against partitioned retrieval. (a) Paired median error, one line per query set and strategy; every line slopes upward. (b) p90 error under both backends on synthetic queries, with one tile width marked.

Three findings follow, and together they invert the premise of the hierarchical design. First, the descriptor ordering is identical under both backends, so it is a property of the descriptors, not of the candidate-set construction; under exhaustive search the two 384-dimensional variants are nearly tied, both within 7 % of the reference of (3), and multi-rotation remains marginally worse than its base, so its deficit persists independently of page selection.

Second, exhaustive search on real imagery works. With the CLS descriptor, the 120 m flight reaches a median of 68.7 m with 60.0 % of frames within one tile width and the 100 m flight 127.5 m, so the apparent failure of Table I is largely a failure of the coarse stage. The ordering across flights is consistent with Section IV-B: the 120 m flight, whose horizontal scale matches the database to 0.3 %, performs best, although altitude is confounded with flight identity.

Third, the tail is a partitioning artefact. Under exhaustive search, the 90th-percentile (p90) localization error of CLS on synthetic queries is 65.7 m, below one tile width, compared with 1897.5 m under partitioning; the catastrophic-failure rate of mean-patch likewise falls from 22.0 % to 1.0 %. The long tail in Table I is therefore largely associated with queries whose coarse stage excludes the page holding the exhaustive top-1 candidate, rather than with fine-stage errors. Coarse recall in Table III quantifies this loss, since the coarse stage discards roughly a quarter of the exhaustive answers on synthetic queries and 29 to 43 % on real frames. Section VI-B interprets the asymmetry between strategies.

Against this cost, Table IV shows what partitioning returns, a larger index that is also slower on the edge device. Workstation warm-cache figures show partitioning ahead by

under 0.1 ms for CLS, irrelevant beside a 136 ms query and inverted on the target hardware. For 3213 vectors, exhaustive inner-product search is not a bottleneck worth optimizing.

TABLE IV

PER-QUERY COST ON THE EDGE DEVICE, MEASURED ON THE SYNTHETIC QUERY SET. REAL-FLIGHT COSTS, WHICH ARE DOMINATED BY VIDEO DECODING, ARE REPORTED IN SECTION V-D.

Configuration	MiB	Search [ms]	Inference [ms]	Total [ms]	Energy [J]
CLS, flat	4.73	0.404	30.43	49.06	1.034
CLS, partitioned	5.50	0.792	—	—	—
radial, flat	37.68	3.389	34.01	55.75	1.103
radial, partitioned	38.50	3.407	—	—	—
multi-rotation, part.	38.50	—	125.94	154.32	2.701

C. Partition granularity and cache

Comparing page sizes at fixed k_p confounds granularity with search budget, so k_p was swept per configuration at matched ρ of (5). At the evaluated matched coverage levels, finer pages perform better, consistent with the page descriptor of (9) becoming less discriminative as a page grows more heterogeneous. At $P = 2048$ m a page holds a median of 637 tiles over 4.2 km² and its centroid lies close to the map-wide mean, whereas at $P = 600$ m it holds 56 tiles and coarse recall rises from 0.760 to 0.960 at the same ρ , the median error falling from 42.26 to 33.89 m. The worst-to-best ratio at $\rho \approx 50\%$ is 1.25 for CLS (95 % CI [1.12, 1.33]) and 1.35 for radial (CI [1.15, 1.61]). The coarse stage is therefore not inherently unworkable, only configured too coarsely; even at its best it converges toward exhaustive search (32.60 m) rather than improving on it.

Cache capacity affects only page loading, never retrieval output, and median error is invariant across the sweep. Page loading dominates whenever $\kappa > 1$, reaching 96 % of the 7.74 ms search time at $\kappa = 4$ with 63 pages, against 0.38 ms at $\kappa = 1$; since the entire CLS index is 4.73 MiB and fits in memory many times over, the regime in which the cache matters is one this system has no reason to enter.

D. Edge profiling

All figures in this section come from the Jetson platform of Section III-F, over 15 conditions with three repetitions each, 45 runs with no failures, under pinned clocks and float16 inference, with a between-run standard deviation of at most 1.5 ms on totals of 49 to 154 ms. The profiled runs use the partitioned configuration, which differs from the exhaustive one only in the search stage, 0.79 against 0.40 ms, or below 0.3 % of the end-to-end budget, so the timings below apply to either backend. Fig. 5 gives the latency decomposition.

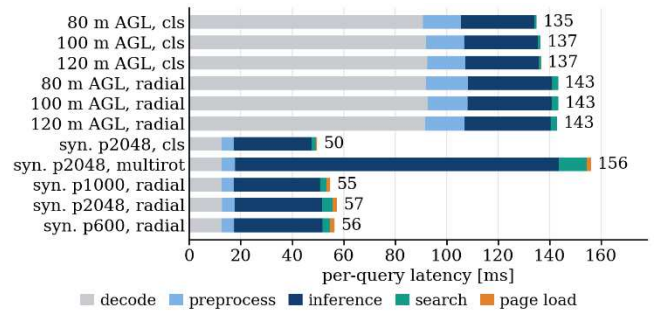


Fig. 5. Per-query latency on the edge device, decomposed into decode, preprocess, inference, search and page load; totals in milliseconds are annotated.

JPEG decoding dominates the real-flight pipeline. Averaged over the three flights, the recommended configuration costs 136 ms per frame at 1.65 J and 12.1 W, of which decoding a 1920×1080 frame from disk accounts for 92 ms, or 67 % of the budget, more than three times the transformer forward pass at 28.6 ms. It therefore runs at 7.4 FPS including decoding and 22.6 FPS without it, so the decode term is a property of this offline stored-video harness rather than of the retrieval method. On synthetic queries, which are decoded from small tiles rather than video frames, decoding falls to 12.5 ms and the same configuration costs 49.1 ms at 1.03 J with a 30.4 ms forward pass, so the average power rises from 12.1 to 21.1 W while the energy per query halves. Table IV reports this condition; per-stage timings are quoted per condition throughout.

Table IV gives the cost of each strategy on the device. Multi-rotation aggregation costs $3.70 \times$ the inference time, $2.77 \times$ the end-to-end latency and $2.45 \times$ the energy of its own base, in exchange for a median error 2.5 % worse, so on the target hardware the most expensive configuration is also the least accurate, and the cheapest the most accurate.

Device-level resource use is moderate, with peak resident memory of 3.3 to 3.6 GB of the platform's 7.4 GB unified memory, dominated by the runtime and model rather than by the index, and GPU utilization peaks at 58 % for multi-rotation. Maximum SoC temperature across all 45 runs was 53 to 59 °C, so every measurement reflects sustained rather than burst performance.

Half-precision inference is effectively free in accuracy terms, since radial and multi-rotation reproduce their workstation median error to six decimal places, and for CLS one query of 200 changed its top-1 tile, moving the median from 42.263 to 42.616 m.

VI. DISCUSSION

Each of the three negative outcomes of Section V traces to a mechanism that follows from the definitions of Section III rather than from the measurements alone. Section VI-A treats the descriptor, Section VI-B the retrieval backend, and Section VI-C what limits real-flight performance. Section VI-D states the limitations.

A. Why radial pooling and multi-rotation do not pay

The radial descriptor was designed on the premise that encoding coarse spatial layout should discriminate better than a single global vector; its definition suggests why the measurements do not support that premise. The bins are equal in radius, not in area, so (6) places 24 tokens in the innermost bin and 248 in the sixth while every bin contributes the same 384 dimensions; and because $q_{\max}$ is the corner distance rather than the inscribed radius, the two outermost bins are supported only on the four corner regions and contribute a quarter of the descriptor, precisely the regions least likely to be shared between a database tile and a query framed or oriented differently.

This contrasts with GeoCluster's centre-favouring spatial prior, which improves retrieval on its evaluated aerial datasets [13]. Equal-dimensional radial annuli have unequal areas and token counts, so the present descriptor allocates disproportionate capacity to peripheral regions. Methods [14], [15] instead learn their spatial treatment during task-specific training. GeoCluster [13] is therefore the closest comparison for a fixed, training-free prior. The rotation stratification of

Section V-A is consistent with a third mechanism, since at intermediate angles the partition itself moves, but every interval contains unity and we do not claim more.

Multi-rotation aggregation increases inference cost $3.70 \times$ and returns a median error 2.5 % worse than its base. The merge rule of (8) provides a plausible explanation: max-over-rotations can raise scores for both correct and incorrect tiles, while the non-rotation-equivariant encoder gives no guarantee that the correct match benefits preferentially. Table III shows that the deficit persists under exhaustive retrieval, excluding page selection as its cause. The 22 versus 4 discordant real-flight successes show that rotation aggregation can help some frames, but heading is unavailable to attribute these cases to specific orientations.

B. Why partitioning does not pay at this scale

Partitioning was intended to bound memory and to accelerate retrieval by restricting the candidate set. Table IV shows it does neither here. The partitioned index is larger than the flat one, because the coarse index and per-page shards add overhead the reduced candidate set never recovers, and at under 5 MiB the flat index leaves nothing to bound. It is also slower on the target device, where both search times are negligible beside the 28.6 ms forward pass and the 91 ms decode.

The accuracy cost is concentrated in the tail, and the mechanism is the page descriptor of (9). At $P = 2048$ m the centroid of 637 unit vectors drawn from heterogeneous terrain lies close to the map-wide mean, carrying little information about which page a query belongs to, and Section V-C confirms this by lifting coarse recall from 0.760 to 0.960 when the page is reduced to 56 tiles. The coarse stage also generalizes worse than the fine stage, coarse recall falling from 0.735–0.760 on synthetic queries to 0.573–0.711 on real frames. This is why coarse recall must not be read as a descriptor ranking, since it is defined per strategy against that strategy's own exhaustive top-1, so radial exceeds CLS on two flights precisely because it reproduces a poorer exhaustive result more faithfully.

The page cache follows the same logic, since capacity matters only when $\kappa > 1$, and since the whole CLS index resides comfortably in memory on the 7.4 GB device, $\kappa \leq 1$ is the only sensible operating point, at which prefetching can only recover latency a correctly sized cache never spends. We therefore report no prefetch results. None of this shows that metric partitioning is unsound in principle; it shows only that no benefit appears at 3213 vectors and 18.29 km^2 .

C. What limits real-flight performance

Exhaustive search with the CLS descriptor localizes the 120 m flight to a median of 68.7 m with 60.0 % of frames within one tile width, a working result from a consumer UAV, but roughly twice the 32.6 m obtained on synthetic queries. The gap is consistent with the three acquisition differences derived in Section IV-B before any result was consulted, although their individual contributions are not isolated by this experiment. Heading is unknown, so every real query is rotated by an uncontrolled angle relative to the north-up database, whereas synthetic queries are bounded at $\pm 30^\circ$; methods that obtain rotation invariance by construction avoid this [5]. Scale is mismatched and altitude-dependent, a real frame covering 25 to 57 % of the tile area, which motivates multi-height benchmarks [7] and the use of flight height as an auxiliary input [28]. And a 16:9 frame resized to a square input

is compressed along one axis by 1.778. The flight ordering supports this reading, since the 120 m flight, whose horizontal scale matches the database to 0.3 %, gives the best exhaustive result, although altitude is confounded with flight identity.

Two further factors are unquantified. Camera attitude relative to nadir is unverified, and the queries are lossily compressed frames of a scene under different illumination and season from the mosaic, a class of environmental domain shift that degrades aerial retrieval and that environment-adaptive networks address [33]. These factors arise from acquisition rather than descriptor definition and are shared across all strategies, allowing paired descriptor comparisons even though absolute real-flight errors grow.

D. Limitations

The study covers a single 18.29 km² region of one landscape type from one orthomosaic acquisition, and terrain self-similarity is a property of the scene, so conclusions about absolute accuracy do not transfer to other regions. Only one backbone and one tiling are evaluated, DINOv3 ViT-S/16 at 512 × 512 px with 1024 px tiles at 512 px stride, so the descriptor comparison is conditional on that configuration.

Ground truth comes from the aircraft’s uncorrected GNSS receiver with no independent survey control, and neither heading nor gimbal telemetry is available, so real queries cannot be de-rotated and camera attitude cannot be verified against nadir. All real imagery comes from a single platform on a single day, and the block bootstrap rests on 56 effectively independent blocks, which is why the intervals of Section V-A are wide. Latency is measured decoding stored JPEGs, 67 % of the end-to-end budget, so a camera-pipeline integration would change the absolute figures though not the relative cost of the descriptors. The edge device has unified memory, so the reported peak is a device-level resident footprint.

VII. FUTURE WORK

The measurements suggest a clear ordering of what would help most, and none of it concerns descriptor structure or index hierarchy. Heading compensation is a high-priority direction, since deriving query orientation from an IMU or magnetometer and de-rotating to map north before descriptor extraction would address a major uncontrolled variable without changing the index, while the invariant-representation route of [5] is an established alternative. Acquisition-matched preprocessing is computationally inexpensive. Padding a 16:9 frame to square rather than stretching it, and resampling to the database scale using the known altitude, remove two of the three quantified mismatches.

Temporal fusion is a promising route to a usable navigation solution. At 1 Hz and approximately 10 m/s, consecutive observations are strongly constrained by motion. A sequence filter over retrieval posteriors could potentially suppress isolated failures and enforce trajectory consistency. Spatially structured matching remains relevant at the observation level [13], while Monte-Carlo localization provides a framework for combining image-to-map matching with motion information over time [34]. Its accuracy in the present setting requires direct evaluation.

Two further directions follow from the limitations. Tile-centre estimation constrains the attainable error distribution according to (3); geometric refinement against the retrieved tile, as in coarse-to-fine systems [3], [21], could reduce error beyond this quantization constraint. The backbone is frozen

[8], [9], making parameter-efficient adaptation the natural next step. AdaptGeo [12] shows that a lightweight adapter can outperform full fine-tuning under its cross-view protocol. A controlled comparison here should hold descriptor dimensionality, preprocessing and retrieval fixed while testing frozen, adapter-based and selectively fine-tuned models. Recent VPR methods illustrate alternative aggregation and representation designs [17], [19], [30]. The scale at which metric partitioning becomes worthwhile is not identified here and requires evaluation on larger maps.

VIII. CONCLUSION

This work asked whether spatially structured token pooling and hierarchical metric partitioning improve UAV visual geolocalization on edge hardware. Under the evaluated configuration, neither does. Radial token pooling costs eight times the descriptor dimensionality and is significantly worse than both 384-dimensional global baselines on both query sets; a mean-pooling control at the baseline dimensionality shows that the deficit is specific to the evaluated radial construction rather than to patch-token use alone. Multi-rotation aggregation costs $3.70 \times$ the inference time and $2.45 \times$ the energy of its base and returns no accuracy benefit, including under exhaustive search. Hierarchical partitioning costs a factor of 1.3 to 8.4 in median error, occupies 16 % more memory than a flat index, and is $1.96 \times$ slower on the target device.

A 384-dimensional CLS descriptor in a 4.73 MiB flat exact index achieved a 32.60 m median error on synthetic queries and 68.7 m on the 120 m real-flight sequence. On the Jetson Orin Nano, the real-flight pipeline required approximately 136 ms and 1.65 J per frame, with JPEG decoding accounting for about 68% of total latency.

For a map of this size, the results favour a plain global descriptor and an exact flat index; engineering effort is better directed toward heading compensation, acquisition-matched preprocessing and temporal fusion than toward descriptor structure or index hierarchy. Making these effects measurable, attributable to specific design choices, and repeatable on released data and code is the contribution of this work.

AUTHOR CONTRIBUTIONS

Ivan Suchý: Conceptualization; Data curation; Formal analysis; Investigation; Software; Validation; Visualization; Writing - original draft; Writing - review & editing.

Michal Turčaník: Conceptualization; Data curation; Formal analysis; Funding acquisition; Investigation; Methodology; Project administration; Resources; Software. Supervision; Validation; Visualization. Writing - original draft; Writing - review & editing.

ACKNOWLEDGEMENTS

This work is supported by the Ministry of Defence of the Slovak Republic through the inter-ministerial sub-program 06E0I-Research and development in support of state defence.

FUNDING

This work was funded by the Ministry of Defence of the Slovak Republic.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest and they have no known financial interests or personal relationships that could influence the work reported in this paper.

DATA AVAILABILITY STATEMENT

The data and code supporting the findings of this study are publicly available. The processing pipeline, the tile catalogues for all three partition configurations, the synthetic query parameters and the ground-truth positions of the three flights are available in the GitHub repository at <https://github.com/isuchy/uav-geolocalization-spatial-structure>. The reference orthophoto extract, the generated synthetic query set and the original UAV video recordings with their SRT telemetry are archived at <https://doi.org/10.5281/zenodo.23021010>. The flight recordings were captured for this study and are not part of any public benchmark. The reference map is derived from Ortofotozoaika SR, Stred 2024 (RGB), a public product of GKÚ Bratislava and NLC Zvolen, published by ÚGKK SR. The 3213 map tiles are not distributed as image files; each is a fixed raster window into the orthophoto, defined by the tile catalogues, and is read at run time.

REFERENCES

- [1] A. Couturier and M. A. Akhloufi, "A review on absolute visual localization for UAV," *Robotics and Autonomous Systems*, vol. 135, p. 103666, Jan. 2021, doi: 10.1016/j.robot.2020.103666.
- [2] Q. Zhou, H. Tang, Z. Zhang, Y. Xu, F. Xiao, and Y. Jia, "A Hierarchical Absolute Visual Localization System for Low-Altitude Drones in GNSS-Denied Environments," *Remote Sensing*, vol. 17, no. 20, p. 3470, Oct. 2025, doi: 10.3390/rs17203470.
- [3] Q. Ye, J. Luo, and Y. Lin, "A coarse-to-fine visual geo-localization method for GNSS-denied UAV with oblique-view imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 212, pp. 306–322, May 2024, doi: 10.1016/j.isprsjrs.2024.05.006.
- [4] Y. Chen and J. Jiang, "An Oblique-Robust Absolute Visual Localization Method for GPS-Denied UAV With Satellite Imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–13, 2024, doi: 10.1109/tgrs.2023.3342142.
- [5] Y. Liu, J. Bai, Z. Xiao, Y. Lian, and L. Jiao, "A Rotation and Scale Invariant Map Matching Method for UAV Visual Geolocalization," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 19, pp. 1616–1627, 2026, doi: 10.1109/jstars.2025.3639900.
- [6] Z. Zheng, Y. Wei, and Y. Yang, "University-1652: A Multi-view Multi-source Benchmark for Drone-based Geo-localization," *Proceedings of the 28th ACM International Conference on Multimedia*, Oct. 2020, doi: 10.1145/3394171.3413896.
- [7] R. Zhu, L. Yin, M. Yang, F. Wu, Y. Yang, and W. Hu, "SUES-200: A Multi-Height Multi-Scene Cross-View Image Benchmark Across Drone and Satellite," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 9, pp. 4825–4839, Sept. 2023, doi: 10.1109/tcsvt.2023.3249204.
- [8] M. Oquab *et al.*, "DINOv2: Learning Robust Visual Features Without Supervision," *Transactions on Machine Learning Research*, 2024, [Online]. Available: <https://mlanthyology.org/tmlr/2024/oquab2024tmlr-dinov2/>
- [9] O. Siméoni *et al.*, "DINOv3," *Transactions on Machine Learning Research*, 2026, [Online]. Available: <https://mlanthyology.org/tmlr/2026/simeoni2026tmlr-dinov3/>
- [10] Nikhil Keetha *et al.*, "AnyLoc: Towards Universal Visual Place Recognition," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1286–1293, Dec. 2023, doi: 10.1109/lra.2023.3343602.
- [11] X. Meng *et al.*, "AirGeoNet: A Map-Guided Visual Geo-Localization Approach for Aerial Vehicles," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–14, 2024, doi: 10.1109/tgrs.2024.3456912.
- [12] C. Yin, J. Luo, Q. Ye, and X. Zhang, "AdaptGeo: Parameter-Efficient Cross-View Geo-Localization via Frozen Foundation Model and Transformer Adapter," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 63, pp. 1–19, 2025, doi: 10.1109/tgrs.2025.3635418.
- [13] C. Chen, M. He, J. Wang, and Z. Meng, "GeoCluster: Enhancing Visual Place Recognition in Spatial Domain on Aerial Vehicle Platforms," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 3013–3020, Mar. 2024, doi: 10.1109/lra.2024.3363536.
- [14] Q. Chen *et al.*, "SDPL: Shifting-Dense Partition Learning for UAV-View Geo-Localization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 11, pp. 11810–11824, Nov. 2024, doi: 10.1109/tcsvt.2024.3424196.
- [15] H. Lv *et al.*, "Direction-Guided Multiscale Feature Fusion Network for Geo-Localization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–13, 2024, doi: 10.1109/tgrs.2024.3396912.
- [16] F. Radenovic, G. Tolias, and O. Chum, "Fine-Tuning CNN Image Retrieval with No Human Annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1655–1668, July 2019, doi: 10.1109/tpami.2018.2846566.
- [17] S. Izquierdo and J. Civera, "Optimal Transport Aggregation for Visual Place Recognition," *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17658–17668, 2024, doi: 10.1109/cvpr52733.2024.01672.
- [18] F. Lu *et al.*, "SuperVLAD: Compact and Robust Image Descriptors for Visual Place Recognition," *Advances in Neural Information Processing Systems 37*, pp. 5789–5816, 2024, doi: 10.52202/079017-0188.
- [19] A. Ali-bey, B. Chaib-draa and P. Giguère, "BoQ: A Place is Worth a Bag of Learnable Queries," *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17794–17803, 2024, doi: 10.1109/cvpr52733.2024.01685.
- [20] T. Jin, F. Lu, S. Hu, C. Yuan, and Y. Liu, "EDTformer: An Efficient Decoder Transformer for Visual Place Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 35, no. 9, pp. 8835–8848, Sept. 2025, doi: 10.1109/tcsvt.2025.3559084.
- [21] M. He, J. Liu, P. Gu, and Z. Meng, "Leveraging Map Retrieval and Alignment for Robust UAV Visual Geo-Localization," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–13, 2024, doi: 10.1109/tim.2024.3418097.
- [22] I. Lee *et al.*, "Georeferenced UAV Localization in Mountainous Terrain Under GNSS-Denied Conditions," *Drones*, vol. 9, no. 10, p. 709, Oct. 2025, doi: 10.3390/drones9100709.
- [23] F. Yao, C. Lan, L. Wang, H. Wan, T. Gao, and Z. Wei, "GNSS-denied geolocalization of UAVs using terrain-weighted constraint optimization," *International Journal of Applied Earth Observation and Geoinformation*, vol. 135, p. 104277, Dec. 2024, doi: 10.1016/j.jag.2024.104277.
- [24] F. Deuser, K. Habel, and N. Oswald, "Sample4Geo: Hard Negative Sampling For Cross-View Geo-Localisation," *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16801–16810, Oct. 2023, doi: 10.1109/iccv51070.2023.01545.
- [25] J. Sun, H. Sun, L. Lei, K. Ji, and G. Kuang, "TirSA: A Three Stage Approach for UAV-Satellite Cross-View Geo-Localization Based on Self-Supervised Feature Enhancement," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 9, pp. 7882–7895, Sept. 2024, doi: 10.1109/tcsvt.2024.3382717.
- [26] X. Liu, Z. Wang, Y. Wu, and Q. Miao, "SeGCN: A Semantic-Aware Graph Convolutional Network for UAV Geo-Localization," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 6055–6066, 2024, doi: 10.1109/jstars.2024.3370612.
- [27] H. Li, C. Xu, W. Yang, H. Yu, and G.-S. Xia, "Learning Cross-View Visual Geo-Localization Without Ground Truth," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–17, 2024, doi: 10.1109/tgrs.2024.3425608.
- [28] R. Zhu, M. Yang, L. Yin, F. Wu, and Y. Yang, "UAV's Status Is Worth Considering: A Fusion Representations Matching Method for Geo-Localization," *Sensors*, vol. 23, no. 2, p. 720, Jan. 2023, doi: 10.3390/s23020720.
- [29] T. Darcet, M. Oquab, J. Mairal, and P. Bojanowski, "Vision Transformers Need Registers," in *International Conference on Learning Representations*, 2024. [Online]. Available: <https://mlanthyology.org/iclr/2024/darcet2024iclr-vision/>
- [30] F. Lu, X. Lan, L. Zhang, D. Jiang, Y. Wang, and C. Yuan, "CricaVPR: Cross-Image Correlation-Aware Representation Learning for Visual Place Recognition," *2024 IEEE/CVF Conference on Computer Vision*

and *Pattern Recognition (CVPR)*, pp. 16772–16782, June 2024, doi: 10.1109/cvpr52733.2024.01587.

- [31] J. Johnson, M. Douze, and H. Jegou, “Billion-scale similarity search with GPUs,” in *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535–547, 1 July 2021, doi: 10.1109/TBDATA.2019.2921572.
- [32] Geodetický a kartografický ústav Bratislava (GKÚ), “Ortofotomozaika SR, 3. cyklus (2023-2025), Stred 2024 (RGB) [Orthophoto mosaic of the Slovak Republic, 3rd cycle (2023-2025), Central Slovakia 2024 (RGB)].” GKÚ, Bratislava, Slovakia, 2024. [Online]. Available: <https://www.skgeodesy.sk/gku/produkty-sluzby/nastiahnutie/zbgis.html>.
- [33] T. Wang, Z. Zheng, Y. Sun, C. Yan, Y. Yang, and T.-S. Chua, “Multiple-environment Self-adaptive Network for aerial-view geolocalization,” *Pattern Recognition*, vol. 152, p. 110363, Aug. 2024, doi: 10.1016/j.patcog.2024.110363.
- [34] J. Kinnari, F. Verdoja, and V. Kyrki, “GNSS-denied geolocalization of UAVs by visual matching of onboard camera images with orthophotos,” *2021 20th International Conference on Advanced Robotics (ICAR)*, pp. 555–562, Dec. 2021, doi: 10.1109/icar53236.2021.9659333.