drones

MDPI

Article

TRACE-Loc: Causal Trajectory-Constrained Cross-View Localization for GNSS-Denied UAVs

Shouyi Liao *, Haohong Chen, Zehao Li and Zhiguo Liu

School of Intelligence and Control Engineering, Rocket Force University of Engineering, Xi'an 710025, China; chh_201225@163.com (H.C.); wsgfs2013@163.com (Z.L.); lzgc@163.com (Z.L.)
* Correspondence: lsy_nudt@163.com

Highlights

What are the main findings?

- TRACE-Loc decomposes large-area UAV-to-satellite localization into online candidate generation and causal trajectory selection; its recursion without historical projection is a beam-width-limited Viterbi decode. The full reranker gains +9.66 ± 2.69 pp C@1 over single-frame retrieval across three training runs, and the historical-projection stage itself contributes +8.07 ± 3.75 pp over the identical recursion without projection.
- Heading, elapsed time, and visual speed jointly constrain candidate paths, preventing transient single-frame mismatches from directly changing the location sequence.

What are the implications of the main findings?

- Candidate-level temporal inference provides a lightweight online mechanism for obtaining an initial fix without a reliable prior position; its early-acquisition advantage over single-frame retrieval is concentrated at the start of the cold start, whereas the particle filter remains preferable for long-horizon recovery.
- The reranker is compatible with different cross-view retrieval frontends, allowing descriptor design and trajectory-consistency modeling to be evaluated separately.

Abstract

UAV-to-satellite visual localization can provide absolute position observations when Global Navigation Satellite System (GNSS) signals are denied or lost. Without a reliable initial position, however, cross-view differences and repetitive textures can cause single-frame retrieval to jump to distant but visually similar regions. This paper presents TRACE-Loc, an online trajectory-constrained cross-view retrieval framework for initial localization. A DINOv3 frontend with view-conditioned low-rank adaptation (VC-LoRA) generates current-frame candidates from a global satellite gallery. The online reranker projects recent location hypotheses using heading, elapsed time, and causal visual speed, then recursively selects a motion-consistent path from the union of current retrievals and historical projections. The path recursion without historical projection coincides with a beam-width-limited Viterbi decode over the retrieval candidates, so the method can be read as beam Viterbi plus causal candidate projection. Experiments use a unified gallery of 15,130 UAV-VisLoc satellite tiles and 1215 queries from two regions excluded from both training and parameter selection. Across three independent training runs, TRACE-Loc raises C@1 from 48.67 ± 0.41% to 58.33 ± 2.84%, a paired gain of 9.66 ± 2.69 percentage points; the single representative run used for detailed comparisons increases C@1 from 49.05% to 61.23%, reduces median localization error from 150.92 m to 93.75 m, and raises D < 300 m from 53.66% to 70.62%. The same reranker improves C@1 across six retrieval frontends. In reset-at-every-start cold-start evaluation, TRACE-Loc increases the Kaplan–Meier probability of obtaining a stable fix

Check for updates

Academic Editor: Giordano Teza

Received: 20 August 2026
Revised: 2 October 2026
Accepted: 3 October 2026
Published: 22 October 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

within 30 s over single-frame retrieval by 2.33, 9.27, and 5.89 percentage points at the 50, 100, and 300 m thresholds (paired block-bootstrap 95% CIs excluding zero at every threshold), while the particle filter attains statistically indistinguishable early-fix probabilities and recovers faster over longer observation periods; the reported gains therefore concern acquisition from an empty state relative to single-frame retrieval, not an advantage over the particle filter. These results show that candidate-level trajectory constraints improve framewise localization in a large gallery with millisecond-level reranking overhead and increase the probability of an early stable fix.

Keywords: UAV geo-localization; cross-view image retrieval; satellite imagery; GNSS-denied navigation; trajectory-constrained re-ranking; visual odometry

1. Introduction

Unmanned aerial vehicles (UAVs) are widely used for inspection, mapping, search and rescue, and low-altitude reconnaissance, all of which depend on reliable absolute positioning. Outdoors, UAVs typically rely on the Global Navigation Satellite System (GNSS), yet GNSS signals are vulnerable to blockage, interference, spoofing, and deliberate denial. When GNSS becomes unreliable, navigation must be maintained with onboard sensors and prior maps. Visual or visual–inertial odometry provides continuous relative motion, but its error accumulates over time. UAV-to-satellite visual geolocalization instead matches onboard images to a georeferenced orthophoto map and can directly recover a global position. This study focuses on rapidly obtaining an absolute position after GNSS loss when the initial position is unknown or unreliable, a process referred to here as *initial localization* from an empty state.

Initial UAV-to-satellite localization presents three main difficulties. *(i) Large-area global search.* Without a reliable prior position, matching cannot be restricted to a local map. Applying local correspondence or geometric verification independently to every satellite tile is incompatible with rapid online localization. *(ii) Cross-view domain differences.* UAV images and orthorectified satellite tiles differ substantially in altitude, viewpoint, resolution, orientation, illumination, and texture statistics, which limits the direct transfer of general visual place recognition (VPR) models to UAV-to-satellite imagery. *(iii) Framewise instability in weak or repetitive textures.* Weakly textured regions such as open water contain few distinctive structures, whereas farmland and coastlines often produce several similar candidates. Directly accepting the single-frame top-1 retrieval can therefore cause large jumps on the map and does not provide a stable initial fix.

Existing work follows two broad directions. The first improves cross-view retrieval through stronger global descriptors, cross-view contrastive learning, or foundation-model adaptation [1–5]. These methods scale to large galleries, but their final position is usually still determined by framewise top-1 retrieval and remains sensitive to transient mismatches in weak or repetitive textures. The second direction introduces geometric or temporal constraints, including local feature matching, coarse-to-fine registration, sequence matching, and filter-based localization [6–12]. Such methods improve consistency, but they often require a useful initial position, a longer observation window, or substantial per-candidate matching cost, which does not fully match the requirements of rapid initial localization.

To address this setting, we propose **TRACE-Loc** (**TR**ajectory-**A**ware **C**ross-view r**E**trieval **Loc**alization). TRACE-Loc does not change the underlying cross-view retrieval task; instead, it treats retrieval results as candidate observations for online localization rather than as final framewise decisions. Each frame is still queried independently against the complete

satellite gallery. A candidate-level state update then combines current visual similarity with motion relationships between adjacent time steps to select a location consistent with the flight trajectory. Unlike temporal averaging of similarity scores or retrieval ranks, this update does not wait for future images to align a complete sequence and therefore supports causal framewise localization without a known initial position.

The TRACE-Loc online update consists of three connected steps. First, a cross-view retrieval frontend queries the heading-aligned UAV image against the global gallery and returns the top-K_{ret} candidates. Recent low-cost states are then projected to the current time using heading, elapsed time, and visual speed and merged with the current retrieval candidates. Finally, node confidence and motion consistency update a bounded set of candidate paths, and the satellite tile at the end of the lowest-cost path is returned. Historical projection keeps a recent location hypothesis available when it temporarily leaves the current retrieval list, thereby separating a single-frame retrieval error from the continuous localization output. This work addresses initial candidate localization in a large gallery and reports discrete tile centers; a subsequent local geometric module can refine a stable candidate to continuous coordinates.

On the unified UAV-VisLoc global gallery, candidate-level trajectory constraints improve framewise localization for several retrieval frontends. Reset-at-every-start cold-start evaluation further separates early position acquisition from long-term recovery. Compared with single-frame retrieval, TRACE-Loc raises the Kaplan–Meier probability of a stable fix within 30 s by 2.33, 9.27, and 5.89 percentage points at the 50, 100, and 300 m thresholds (paired bootstrap 95% CIs excluding zero at every threshold), whereas the particle filter achieves comparable early-acquisition probabilities and shorter waiting times over longer observation periods; our results therefore do not establish an advantage over the particle filter. This difference indicates that rapid candidate acquisition and the continued elimination of incorrect historical hypotheses are distinct stages of initial localization.

The main contributions are as follows:

1. We formulate large-area UAV-to-satellite localization without a reliable initial position as causal path selection over a global candidate set and introduce the TRACE-Loc online trajectory reranking framework. Cross-view retrieval generates candidates, while visual similarity and short-horizon motion consistency jointly determine the current position. The recursion without historical projection is a beam-width-limited Viterbi decode over the retrieval candidates; the contribution of TRACE-Loc is therefore isolated to the causal historical candidate projection stage, which we quantify explicitly.
2. We introduce a motion-prior-guided historical candidate projection mechanism. Recent location hypotheses are propagated using heading, elapsed time, and strictly causal visual speed, expanding the available state set when single-frame retrieval temporarily misses a useful location without requiring a predefined local search region. Projection raises C@1 by 11.93 percentage points over the identical recursion without projection (61.23% vs. 49.30%), correcting 149 outputs while harming 4.
3. We establish a reset-at-every-start first-stable-fix protocol with censored observations to evaluate early position acquisition separately from long-term recovery. The protocol reveals that historical projection increases the probability of an early stable fix relative to single-frame retrieval but that incorrect historical hypotheses can prolong recovery for difficult starts.

2. Related Work

2.1. Cross-View Retrieval and Visual Representation

UAV-to-satellite geolocalization is commonly formulated as cross-view visual place recognition: a UAV image queries a geotagged satellite or orthophoto gallery for the same

region. Early studies already used onboard-to-satellite image matching for GNSS-denied localization [13,14]. More recent benchmarks and models, including University-1652, DenseUAV, TransGeo, and UAV-VisLoc, have advanced UAV-to-satellite retrieval [15–18]. These datasets progressively cover changes in altitude and camera attitude as well as larger satellite galleries, shifting the task from ideal one-to-one matching toward candidate search under realistic image overlap. In UAV-VisLoc, for example, a UAV image can cover several adjacent tiles. One-to-many retrieval reconstructs positive relations and the training loss to reduce conflicts caused by incomplete labels [19]. MMGeo extends cross-view localization to settings with multimodal cues by incorporating depth, point-cloud, and text queries in addition to images [20].

The central objective in this direction is an image representation robust to changes in viewpoint, scale, and appearance. NetVLAD [1] introduced learnable aggregation for global descriptors. MixVPR [2], SALAD [3], and Sample4Geo [4] improve retrieval through feature mixing, aggregation design, and cross-view sampling, respectively. The DINOv2 and DINOv3 vision foundation models [5,21], together with parameter-efficient adaptation methods such as LoRA [22,23], strengthen representation learning when task-specific data are limited. GenGeo combines local DINOv2 features with SALAD aggregation to improve cross-dataset generalization [24]. These studies improve candidate quality through supervision, input modality, and representation design, but the current-frame ranking usually still determines the localization result. As the gallery expands, or as farmland, roads, and building clusters exhibit repetitive textures, one high-confidence mismatch can move the estimated position to a distant similar region. Improving a descriptor and stably selecting a sequence of candidates are therefore related but distinct problems.

Figure 1 places cross-view retrieval together with geometric refinement, sequence matching, and recursive state estimation in a common technical context and summarizes their shared gap under global initial-localization conditions.

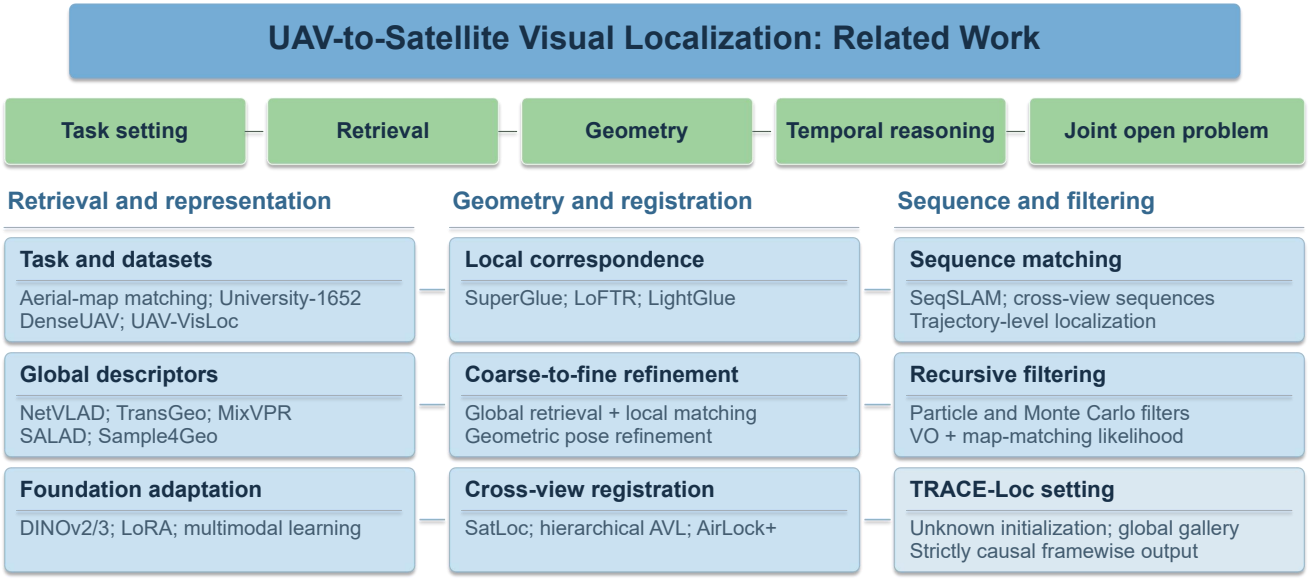


Figure 1. Technical context of UAV-to-satellite visual localization, spanning cross-view retrieval, geometric refinement, sequence matching, and recursive state estimation. TRACE-Loc addresses causal candidate selection under global unknown initialization.

2.2. Geometric Matching and Local Position Refinement

Beyond global-descriptor retrieval, image-to-map systems often use local correspondence and geometric constraints to improve localization accuracy. Learned local matchers

such as SuperGlue, LoFTR, and LightGlue [6,7,25] establish dense or semi-dense correspondences between candidate image pairs. Coarse-to-fine localization systems combine global retrieval, local matching, and geometric estimation to refine pose within a candidate region [8]. The hierarchical absolute visual localization system of Zhou et al. first performs global-retrieval initialization and then registers images within a sliding submap bounded by the previous frame while using an IMU to correct image orientation [26]. AirLock+ combines cross-view matching, homography estimation, and pixel-to-geographic mapping for continuous UAV target geolocation [27]. These systems show that local geometry can provide observations finer than a tile center when the candidate region is already known and correspondences are sufficiently stable.

UAV-to-satellite geometry is less controlled than conventional same-view matching. Oblique onboard imagery and orthorectified tiles differ in scale, local planarity, and visibility, which can prevent single-frame local matches from supporting a stable global transformation. More importantly, geometric refinement generally operates after retrieval has already identified the correct neighborhood. If initialization selects a distant but similar region, subsequent local registration does not naturally recover the global search. Applying local matching and RANSAC to many candidates during large-area initialization also increases online computation. Geometric refinement is therefore well suited to recovering sub-tile positions after a reliable candidate has been obtained, whereas the stabilization of global candidates over consecutive observations remains a separate problem.

2.3. Sequence Matching and Recursive State Estimation

Temporal consistency is a classical means of reducing single-frame ambiguity. SeqSLAM [9] showed that image sequences support more reliable place recognition under substantial appearance change. Subsequent studies applied dynamic time warping and sequence reranking to cross-season and cross-view settings [28], including direct UAV-to-satellite image sequence matching [10]. These approaches suppress incidental mismatches through multi-frame context, but they usually require a longer bidirectional context or a structured reference route and are therefore better suited to offline sequence alignment.

NaviLoc recently reformulated single-frame cross-view retrieval as trajectory-level localization [29]. It uses a relative trajectory from visual–inertial odometry to search for a global SE(2) alignment over the complete observation sequence, followed by local-window correction and trajectory smoothing that fuse retrieval anchors. This result shows that trajectory structure can recover a consistent global solution from many single-frame mismatches. Trajectory-level alignment and the online initial-localization problem studied here, however, impose different output conditions. The former optimizes an accumulated trajectory segment, whereas the latter starts from an empty state, updates the position using only current and historical observations as each frame arrives, and explicitly measures the time required to form the first stable fix.

A second family treats visual matching results as observation likelihoods and recursively estimates the global position through Bayesian filtering. Particle filters and Monte Carlo localization combine visual odometry with orthophoto matching scores [11–13] and can gradually converge from a weak prior. Their efficiency and stability depend on the particle count, motion model, and map-matching cost at each update. Overall, single-frame retrieval, local geometry, trajectory alignment, and recursive filtering provide candidate generation, position refinement, long-term consistency, and continuous state estimation, respectively. The role of candidate-level constraints in the first-stable-fix process remains less thoroughly characterized when unknown initialization, a unified global gallery, and causal framewise output are required simultaneously. TRACE-Loc targets this protocol by applying short-horizon motion consistency to online selection among a bounded set of

retrieval candidates and treating first-stable-fix time as an evaluation dimension distinct from single-frame retrieval accuracy.

3. TRACE-Loc Candidate-Level Trajectory Localization

3.1. Overall Pipeline and Problem Definition

Figure 2 presents the complete TRACE-Loc pipeline. Single-frame cross-view retrieval generates candidates from the global satellite gallery but does not directly determine the current position. When each frame arrives, current retrieval candidates are merged with motion-based projections of recent historical states. Transition costs between candidates are jointly determined by heading, elapsed time, and visual speed, and online recursion selects a motion-continuous location sequence. The estimate at each time therefore depends only on current and historical observations and requires neither future images nor a reliable initial position. This decomposition provides a common state interface for candidate generation, historical projection, and online path updates.

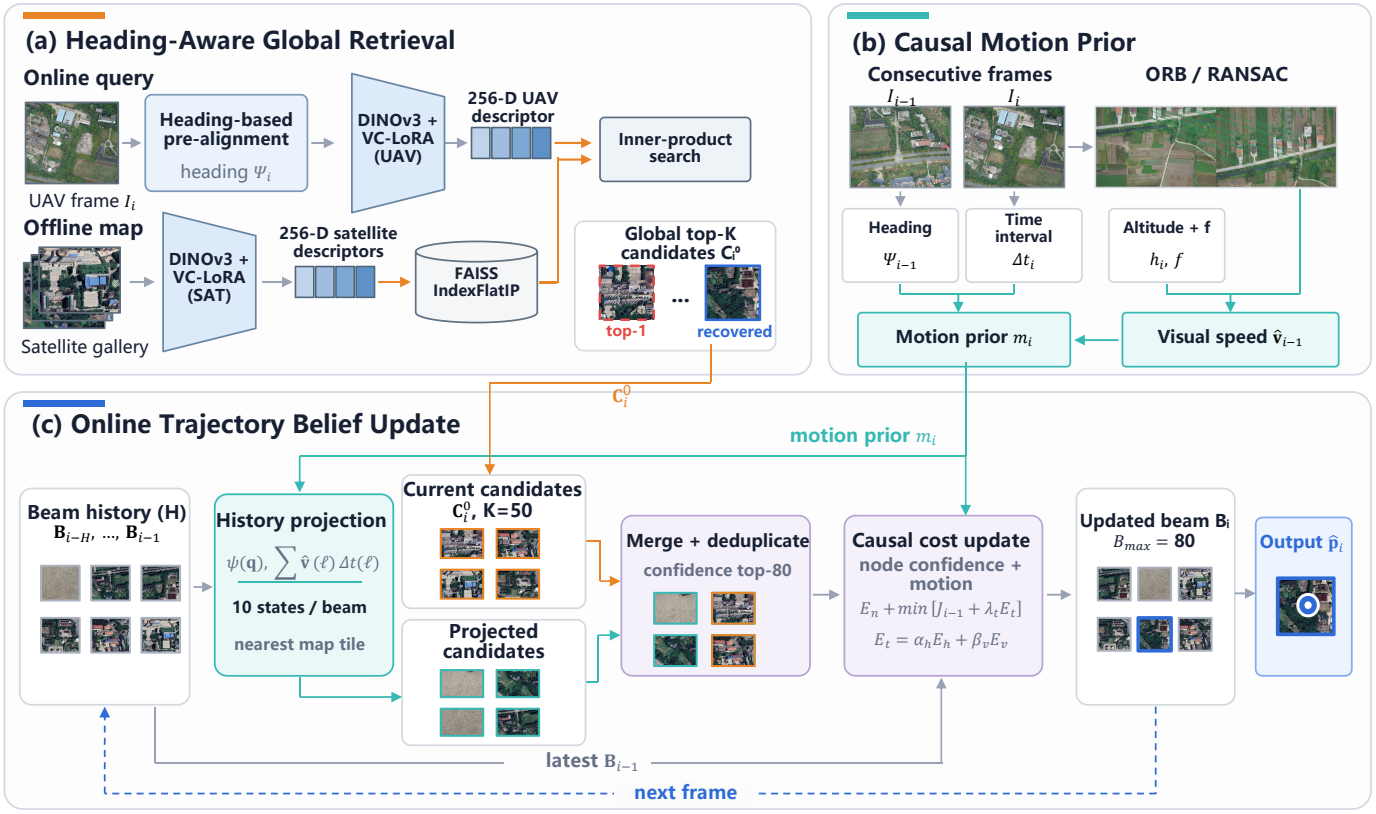


Figure 2. Overview of the TRACE-Loc pipeline. (**a**) Heading-aligned cross-view retrieval generates satellite candidates. (**b**) Adjacent-frame visual speed, heading, and elapsed time define a causal motion prior. (**c**) Current and projected candidates are merged and reranked by visual and motion consistency. The dashed arrow carries the updated beam into the next history window.

At discrete time i, the UAV obtains the observation

$$\mathbf{o}_i = \left(I_i^u, \psi_i, t_i, h_i\right), \qquad \mathcal{O}_{r:i} = \{\mathbf{o}_r, \ldots, \mathbf{o}_i\}, \tag{1}$$

where I_i^u is the onboard image, ψ_i is heading, t_i is the timestamp, h_i is relative height, and r denotes the starting frame of online localization. The unknown true planar position is $p_i^\star \in \mathbb{R}^2$, represented in a local East–North–Up (ENU) coordinate system. Heading is zero toward north and increases clockwise; logged angles are converted to radians before trigonometric operations. The interval between adjacent observations is $\Delta t_i = t_i - t_{i-1}$.

Adjacent onboard images, relative height, and known camera intrinsics $\mathbf{K}_c$ provide the scalar visual-speed estimate $\hat{v}_{i-1}$.

The satellite map is pre-tiled into the gallery

$$\mathcal{G} = \{(T_j, p_j)\}_{j=1}^{N_g}, \tag{2}$$

where T_j is the j-th tile and p_j is its center in the local metric ENU frame. The discrete state space formed by all tile centers is $\mathcal{P}_g = \{p_j\}_{j=1}^{N_g}$. No spatial prior is available at cold-start time r, so the first frame queries the complete gallery $\mathcal{G}$. Subsequent observations recursively update the localization state in the same global state space.

The output at time i is the causal mapping

$$\hat{p}_i = \mathcal{F}_i(\mathcal{G}, \mathbf{K}_c, \mathcal{O}_{r:i}), \qquad \hat{p}_i \in \mathcal{P}_g, \quad i \geq r, \tag{3}$$

which maps the observation prefix available at time i to one satellite-tile center. Online localization without an initial position is thereby expressed as global discrete-state estimation.

If only the visual similarity s_{ij} between the current frame and tile j is considered, the single-frame estimate is

$$\hat{p}_i^{\mathrm{sf}} = p_{\arg\max_j s_{ij}}, \tag{4}$$

which ignores motion between adjacent frames. TRACE-Loc instead treats the top-K_{ret} results of each frame as observation candidates and augments them with projected historical states. Let the extended candidates at time ℓ be $\tilde{\mathcal{C}}_\ell = \{(\tilde{p}_\ell^k, a_\ell^k)\}_{k=1}^{\tilde{K}_\ell}$. A candidate path from r to i is $\boldsymbol{\pi}_{r:i} = (k_r, \ldots, k_i)$, where $k_\ell \in \{1, \ldots, \tilde{K}_\ell\}$. Path selection is formulated as

$$\hat{\boldsymbol{\pi}}_{r:i} = \arg\min_{\boldsymbol{\pi}_{r:i}} \left[\sum_{\ell=r}^{i} E_{\mathrm{node}}(\ell, k_\ell) + \lambda_T \sum_{\ell=r+1}^{i} E_{\mathrm{trans}}(\ell{-}1, k_{\ell-1}, \ell, k_\ell) \right], \tag{5}$$

where the node cost measures the visual confidence of a candidate and the transition cost measures heading and speed consistency between consecutive candidates. For the optimal path $\hat{\boldsymbol{\pi}}_{r:i} = (\hat{k}_r, \ldots, \hat{k}_i)$, the current output is the endpoint of the optimal prefix, $\hat{p}_i = \tilde{p}_i^{\hat{k}_i}$. TRACE-Loc updates Equation (5) online with a bounded beam. Figure 2 summarizes the relationship among heading-aware retrieval, historical candidate projection, and causal belief reranking.

3.2. Heading-Aligned Cross-View Candidate Generation

Single-frame cross-view retrieval serves only to generate candidate positions. Satellite tiles follow a north-up convention, whereas the orientation of a UAV image changes with heading. Each UAV image is first rotated to the north-up convention using its logged heading, then center-cropped to a square and resized to 512×512:

$$\hat{I}_i^u = \mathrm{Crop}\big[\mathcal{R}_{\mathrm{img}}(-\psi_i)\, I_i^u\big], \tag{6}$$

where $\mathcal{R}_{\mathrm{img}}$ is the image rotation operator and $\mathrm{Crop}[\cdot]$ denotes center cropping. This operation corrects the dominant orientation difference between views, while the feature encoder handles the remaining appearance and scale variations.

The default cross-view retrieval frontend combines a frozen DINOv3 ViT-B/16 backbone [5] with VC-LoRA. VC-LoRA adds a shared low-rank residual and a view-conditioned residual to the later backbone blocks. The shared branch learns feature transformations reusable across UAV and satellite imagery, whereas the view-conditioned branch selects a UAV or SAT residual according to the input view. For a linear transformation containing both residual types, the output is

$$\mathbf{y}_\nu = \mathbf{W}_0\mathbf{x} + \frac{\alpha_{\text{lora}}}{\rho}(\mathbf{B}_{\text{sh}}\mathbf{A}_{\text{sh}}\mathbf{x} + \mathbf{B}_\nu\mathbf{A}_\nu\mathbf{x}), \qquad \nu \in \{\text{uav}, \text{sat}\}, \tag{7}$$

where $\mathbf{x} \in \mathbb{R}^{d_{\text{in}}}$, $\mathbf{y}_\nu \in \mathbb{R}^{d_{\text{out}}}$, and $\mathbf{W}_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ is frozen. The matrices $\mathbf{A}_{\text{sh}}, \mathbf{A}_\nu \in \mathbb{R}^{\rho \times d_{\text{in}}}$ and $\mathbf{B}_{\text{sh}}, \mathbf{B}_\nu \in \mathbb{R}^{d_{\text{out}} \times \rho}$ parameterize the low-rank residuals, ρ is their rank, and α_{lora} is the scaling coefficient. The subscript sh denotes parameters shared between views, whereas ν denotes the active input view. Linear transformations that do not require view differentiation retain only the shared residual. In each adapted block, the query–key–value projection, the attention output projection, and both MLP layers receive low-rank residuals; among these, the mlp.fc1 adapters are shared and the mlp.fc2 adapters are view-specific (Section 4 lists the per-block ranks). Figure 3 shows the encoder and descriptor head.

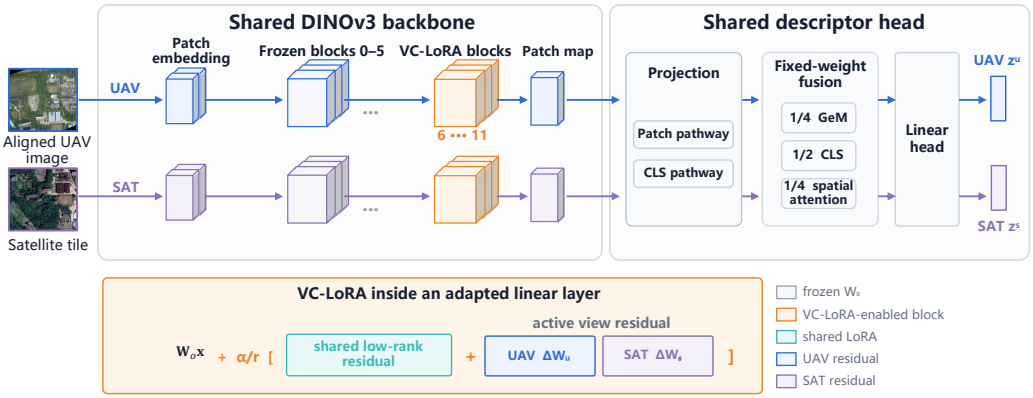


Figure 3. Architecture of the DINOv3/VC-LoRA retrieval frontend. A shared frozen backbone encodes aligned UAV images and satellite tiles, while later blocks add shared and view-specific low-rank residuals. The descriptor head fuses patch and CLS features into a common embedding space.

To retain both local texture and global semantics, the descriptor head projects the patch feature map and CLS token separately into a 256-dimensional space and extracts GeM pooling [30] and spatial-attention pooling from the patch features. GeM, spatial attention, and CLS representations are fused with weights 1/4, 1/4, and 1/2, respectively, so that local structure and global semantics jointly contribute to matching. A shared linear map and L_2 normalization then produce the UAV descriptor $\mathbf{z}_i^u$ and satellite descriptor $\mathbf{z}_j^s$.

Training labels are determined from the spatial relationship between the UAV image footprint and satellite tiles. For subset s, let $c_s = \min(W_s, H_s)$ be the side length of the center square crop and f_s the calibrated focal length. To distinguish label construction from the framewise relative height h_i used for online visual speed, define the label height h_i^{lab} as the logged altitude minus the DEM elevation at the first-frame position of the subset. The square ground-footprint side length of query i is $F_i = c_s h_i^{\text{lab}} / f_s$. Let S_j be the ground side length of tile j. The local equidistant distance between the query and tile centers is

$$d_{ij} = \sqrt{\left[111320\cos\left(\frac{\pi\phi_i}{180}\right)(\lambda_j - \lambda_i)\right]^2 + \left[111320(\phi_j - \phi_i)\right]^2}, \tag{8}$$

where longitude and latitude (λ, ϕ) are measured in degrees and d_{ij} is measured in meters. The spatial relationship between the query footprint and tile is represented by the center-distance overlap proxy

$$\omega_{ij} = \min\left(1, \frac{\max[0, (F_i + S_j)/2 - d_{ij}]}{\min(F_i, S_j)}\right). \tag{9}$$

A tile is labeled as a strong positive when $\omega_{ij} \geq 0.5$, or when its boundary contains the query location and $d_{ij} < 200\,\text{m}$. Tiles with $\omega_{ij} \in [0.2, 0.5)$ are semi-positive, and the remaining tiles within 1000 m of the query form the buffer set. Figure 4 shows examples of the three label types.

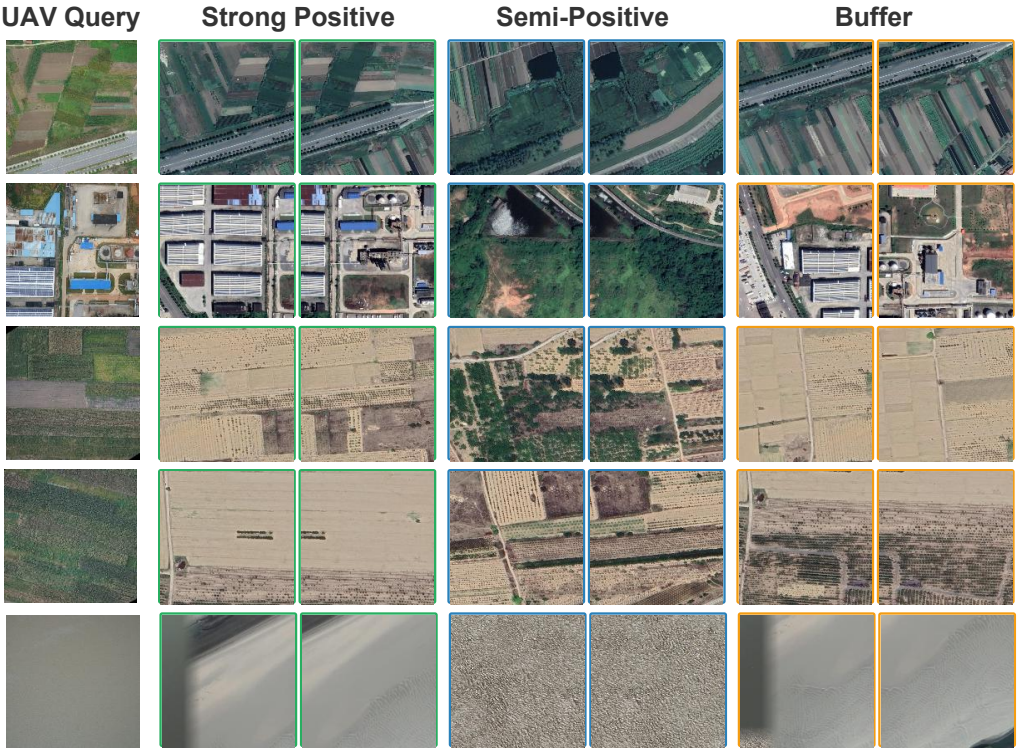


Figure 4. Examples of UAV queries and satellite tiles with different labels. Each row shows one UAV query together with two strong positives, two semi-positives, and two buffer tiles. Green, blue, and orange borders denote the three label types, respectively.

The adapters and descriptor head are trained with a one-way UAV→SAT multi-positive InfoNCE objective. For each query, the two highest-scored strong positives enter the numerator, while semi-positive and buffer samples are excluded from the negatives. The temperature is 0.07.

All gallery descriptors are precomputed and stored in an exact FAISS `IndexFlatIP` . Because the descriptors are normalized, the inner product $s_{ij} = (\mathbf{z}_i^u)^\top \mathbf{z}_j^s$ equals cosine similarity. For each frame, the K_{ret} highest-scoring satellite tiles form the raw candidate set

$$\mathcal{C}_i^0 = \{(p_i^k, s_i^k)\}_{k=1}^{K_{\text{ret}}}, \tag{10}$$

where p_i^k and s_i^k are the location and similarity of the k-th candidate. We use $K_{\text{ret}} = 50$. These candidates form the current-frame observation nodes; the final location is determined by trajectory reranking.

3.3. Historical Candidate Motion Projection

Appearance ambiguity can cause single-frame top-K_{ret} retrieval to temporarily omit the true location, whereas location hypotheses retained in the recent trajectory can supplement the current candidate set. At historical time q, the online reranker retains $\mathcal{B}_q = ((p_q^b, J_q^b))_{b=1}^{B_q}$, where $B_q = |\mathcal{B}_q|$, p_q^b is a tile center, and J_q^b is the cumulative cost of a trajectory ending at that position. Parameter H is the number of historical time steps used in the current update. For example, when $H = 3$, available states are drawn from $\mathcal{B}_{i-3}$,

$\mathcal{B}_{i-2}$, and $\mathcal{B}_{i-1}$. Their candidates are propagated to the current frame i using a short-term constant-heading model. For state b at $q < i$, the projected position is

$$\begin{aligned} \bar{p}^{b}_{q\to i} &= p^{b}_{q} + d_{q\to i}\mathbf{e}(\psi_q), \\ d_{q\to i} &= \sum_{\ell=q}^{i-1} \hat{v}_\ell \left(t_{\ell+1} - t_\ell\right) = \sum_{\ell=q}^{i-1} \hat{v}_\ell \Delta t_{\ell+1}. \end{aligned} \tag{11}$$

Here $\hat{v}_\ell$ is the interval speed estimated from I^u_ℓ and $I^u_{\ell+1}$, with $\Delta t_{\ell+1} = t_{\ell+1} - t_\ell$. The heading unit vector

$$\mathbf{e}(\psi) = [\sin\psi,\ \cos\psi]^\top \tag{12}$$

follows the convention in Section 3.1; its components correspond to east and north in ENU coordinates. Equation (11) uses the source heading ψ_q to determine propagation direction, while visual speed and elapsed time determine propagation distance.

From each historical state set, at most K_{nb} states with the lowest cumulative costs are selected, and each projected position is matched to the nearest gallery-tile center. Relative cumulative cost is converted into the projected candidate node score

$$\bar{a}^{\,b}_{q\to i} = \eta \exp\left(- \frac{J^b_q - \min_{1\le b'\le B_q} J^{b'}_q}{\tau_{\mathrm{proj}}} \right), \tag{13}$$

where η sets the maximum projection score and τ_{proj} controls its decay with relative cumulative cost. Current retrieval candidates and historical projections are merged into $\tilde{\mathcal{C}}_i = \{(\tilde{p}^k_i, a^k_i)\}_{k=1}^{\tilde{K}_i}$. A retrieval candidate uses $a^k_i = s^k_i$, whereas a projected candidate uses $a^k_i = \bar{a}^{\,b}_{q\to i}$. Deduplication is performed by tile index. When several sources map to the same tile, only the largest node score is retained, preventing correlated historical states from accumulating confidence through repeated projection. The top $B_{\max}$ deduplicated candidates are retained by node score. Historical projection only augments the candidate nodes at the current time; state transitions between adjacent times are still handled by the recursion in Section 3.4.

Interval speed is estimated from adjacent onboard images, relative height, and camera intrinsics. To preserve inter-frame geometry, ORB features are extracted from center-cropped images without heading rotation [31], and matches are filtered with a RANSAC partial-affine model [32]. If the original image size is $W_{\mathrm{im}} \times H_{\mathrm{im}}$, the center square has side $c = \min(W_{\mathrm{im}}, H_{\mathrm{im}})$ and upper-left offset $x_0 = (W_{\mathrm{im}} - c)/2$, $y_0 = (H_{\mathrm{im}} - c)/2$. After scaling by $\kappa = 512/c$, the transformed intrinsics are

$$f'_x = \kappa f_x, \quad f'_y = \kappa f_y, \quad c'_x = \kappa(c_x - x_0), \quad c'_y = \kappa(c_y - y_0). \tag{14}$$

Let $\delta\mathbf{u}_\ell = (\delta u_{x,\ell}, \delta u_{y,\ell})^\top$ be the pixel displacement at the image center produced by the partial-affine model, and let $\bar{h}_\ell = (h_\ell + h_{\ell+1})/2$ be the mean relative height of the adjacent frames. Under a local planar approximation for downward-looking imaging, the raw speed is

$$\hat{v}^{\mathrm{raw}}_\ell = \frac{\bar{h}_\ell}{\Delta t_{\ell+1}} \sqrt{ \left(\frac{\delta u_{x,\ell}}{f'_x} \right)^2 + \left(\frac{\delta u_{y,\ell}}{f'_y} \right)^2 }, \qquad \Delta t_{\ell+1} = t_{\ell+1} - t_\ell. \tag{15}$$

The median of valid raw speeds in a trailing window containing the current and four preceding intervals gives $\hat{v}_\ell$. When image matching, geometric estimation, or the height observation is invalid for the current interval, the constant-speed prior v_{fb} is used. A longer observation gap terminates historical propagation and reinitializes the state from the current top-K_{ret} candidates. Section 4 reports the thresholds, camera calibration, and height source.

The constant-heading projection degrades when the platform turns within the retained history. In addition to the fixed-length history analyzed in Section 5.4, we evaluate a simple turn-triggered reset: when the inter-frame heading change $|\psi_i - \psi_{i-1}|$ (wrapped to $[0^\circ, 180^\circ]$) exceeds a threshold, all historical states are cleared before the current update, so projection restarts from the current retrieval list alone. Section 5.5 reports per-frame localization as a function of the body turn rate $|\Delta\psi|/\Delta t$ and the effect of this reset, which we evaluate as an alternative to the validated history length under aggressive maneuvers.

3.4. Causal Recursion over Candidate Paths

Current-frame retrieval provides a direct visual observation, while historical-state projection supplies candidates predicted from short-term motion. After these two sources are merged, localization becomes online path selection over the extended candidate set. The ordered state sequence $\mathcal{B}_i$ stores trajectory hypotheses ending at current candidates and their cumulative costs; it is referred to below as the trajectory belief state. This state is updated only from the current frame and existing history. Algorithm 1 gives the framewise procedure.

Algorithm 1 Online trajectory-constrained candidate reranking.

Require: Current observation $\mathbf{o}_i = (I_i^u, \psi_i, t_i, h_i)$, camera intrinsics $\mathbf{K}_c$, satellite-gallery index $\mathcal{G}$, and recent observations and states retained since starting frame r
Ensure: Current location $\hat{p}_i$ and updated state $\mathcal{B}_i$
if $i > r$ and Δt_i exceeds the reset threshold **then**
  Clear historical states
else if $i > r$ **then**
  Estimate $\hat{v}_{i-1}$ from adjacent raw images, h_{i-1}, h_i, and $\mathbf{K}_c$; use v_{fb} if invalid
end if
Align I_i^u by Equation (6), extract its descriptor, and retrieve the top-K_{ret} candidates $\mathcal{C}_i^0$
$\tilde{\mathcal{C}}_i \leftarrow \mathcal{C}_i^0$
for each available historical state $\mathcal{B}_q$, $q = \max(r, i-H), \ldots, i-1$ **do**
  Select at most K_{nb} states with the lowest cumulative costs
  Project them to frame i by Equation (11) and assign scores by Equation (13)
  Add the projected candidates to $\tilde{\mathcal{C}}_i$
end for
Deduplicate by tile index, retain the largest score for duplicates, and keep the top $B_{\max}$ entries
For each $k \in \tilde{\mathcal{C}}_i$, compute $J_i(k)$ using Equations (16) and (20)
Sort by Equation (22) to obtain $\mathcal{B}_i$
$\hat{p}_i \leftarrow p_i^1$
return $\hat{p}_i, \mathcal{B}_i$

The node cost of candidate k is determined by its score a_i^k:

$$E_{\text{node}}(i,k) = -a_i^k. \tag{16}$$

Current retrieval candidates use image similarity, while historical projections use the projection confidence in Equation (13). For preceding state b and current candidate k, define the planar displacement $\Delta p_i^{b,k} = \tilde{p}_i^k - p_{i-1}^b$. Its direction in ENU coordinates is $\theta_i^{b,k} = \text{atan2}(\Delta E_i^{b,k}, \Delta N_i^{b,k})$, consistent with the heading convention of zero at north and positive clockwise. When a preceding candidate and the current candidate occupy the same tile center, $\Delta p_i^{b,k} = \mathbf{0}$ and $\text{atan2}(0,0) = 0$, which would inject a spurious heading cost $1 - \cos(-\psi_{i-1})$ that depends on the absolute flight heading rather than on any motion inconsistency. We therefore set the heading term to zero for stationary pairs ($\|\Delta p_i^{b,k}\| < 10^{-6}$ m, i.e., identical tile centers); the speed term is retained, so a path that stays on one tile while the

predicted inter-frame displacement exceeds the grid step is still penalized as kinematically inconsistent. Because tile centers are unique but not perfectly regularly spaced (minimum center separation 2.45 m, median 153.7 m), the exact-zero guard is the minimal correction; a coarser variant that also freezes the heading term below half the median grid step (76.9 m) changes about 7% of candidate pairs; at the frozen configuration it reproduces every main metric exactly, and is reported with the ablations in Section 5.4. The heading-deviation cost is

$$E_{\text{head}}(i{-}1,b,i,k) = \begin{cases} 0, & \|\Delta p_i^{b,k}\| < \varepsilon_0, \\ 1-\cos\left(\theta_i^{b,k} - \psi_{i-1}\right), & \text{otherwise}, \end{cases} \tag{17}$$

with $\varepsilon_0 = 10^{-6}$ m, and the discrepancy between candidate displacement and the distance predicted by visual speed is

$$E_{\text{vel}}(i{-}1,b,i,k) = \left(\frac{\|\Delta p_i^{b,k}\|_2 - \hat{v}_{i-1}\Delta t_i}{\sigma_v \Delta t_i + \varepsilon_d} \right)^2, \tag{18}$$

where $\hat{v}_{i-1}$ is the speed from frame $i-1$ to frame i, $\Delta t_i = t_i - t_{i-1}$, σ_v is the speed-deviation scale, and $\varepsilon_d = 10^{-3}$ m prevents division by zero. The numerator and denominator of Equation (18) are therefore both distances, making the cost dimensionless. Heading and speed form the transition cost

$$E_{\text{trans}}(i{-}1,b,i,k) = \alpha_h E_{\text{head}}(i{-}1,b,i,k) + \beta_v E_{\text{vel}}(i{-}1,b,i,k), \tag{19}$$

where α_h and β_v control the relative weights of the two motion constraints.

Cumulative path cost is propagated from preceding states to current candidates. When the preceding belief is empty, including at the sequence start or after reinitialization caused by a long gap, the current node costs directly initialize the state. Otherwise, one causal transition is applied:

$$J_i(k) = \begin{cases} E_{\text{node}}(i,k), & \mathcal{B}_{i-1} = \varnothing, \\ E_{\text{node}}(i,k) + \min\limits_{1 \le b \le B_{i-1}} \left[J_{i-1}^b + \lambda_T E_{\text{trans}}(i{-}1,b,i,k) \right], & \text{otherwise}. \end{cases} \tag{20}$$

Here λ_T balances visual observations and motion consistency. Because J_{i-1} enters Equation (20) unattenuated, an early incorrect path retains its accumulated advantage indefinitely, which is visible as the slow recovery tail of the cold-start analysis in Section 5.3. We therefore also evaluate a discounted variant,

$$J_i(k) = E_{\text{node}}(i,k) + \min_{1 \le b \le B_{i-1}} \left[\gamma\, J_{i-1}^b + \lambda_T E_{\text{trans}}(i{-}1,b,i,k) \right], \tag{21}$$

with $\gamma \in (0,1]$; $\gamma = 1$ recovers the published recursion and remains the frozen main configuration. Section 5.4 reports the effect of $\gamma \in \{0.8, 0.85, 0.9, 0.95\}$ on framewise accuracy. Section 3.3 has already limited the merged candidates by node score to $\tilde{K}_i \le B_{\max}$. The updated states are then sorted in ascending cumulative cost to form the belief used for the current output and subsequent projection:

$$\mathcal{B}_i = ((p_i^b, J_i^b))_{b=1}^{B_i} = \text{Sort}_{J_i}^{\uparrow}\left(\left\{ (\tilde{p}_i^k, J_i(k)) \right\}_{k=1}^{\tilde{K}_i} \right), \qquad B_i = \tilde{K}_i \le B_{\max}. \tag{22}$$

The current estimate is the first sorted state, $\hat{p}_i = p_i^1$. The first-frame output is equivalent to selecting the candidate with the largest node score; subsequent outputs use both visual observations and inter-frame transitions.

4. Data and Experimental Design

The test sequences are replayed chronologically, and every query searches the same satellite gallery containing all regions. The single-frame model produces a candidate set for each image, whereas TRACE-Loc updates a trajectory state over successive candidate sets. The experiments therefore examine not only whether the current frame is matched correctly, but also whether localization recovers after a mismatch and how quickly a stable position emerges from an empty state.

4.1. Data Split and Global Retrieval Scope

Experiments are conducted on UAV-VisLoc [18], which provides UAV images, regional satellite maps, and flight-log metadata (position, altitude, heading, and timestamp) for 11 geographically disjoint areas in China. We use a subset-holdout split. Subsets 01, 03, 04, 09, and 11, together with 733 frames from subset 08, form the training set (4395 frames). Subsets 05 and 06 and the remaining 300 frames from subset 08 form the validation set (1024 frames). Subsets 02 and 10 are reserved for testing (1215 frames). The training and validation frames from subset 08 do not overlap, although they share the same regional satellite map; neither test region is used for model training or parameter selection.

To examine performance over different surface types, the 1071 frames in subset 02 are divided by their predominant terrain into 337 open-water frames and 734 land frames. Results for these two groups are reported separately.

GeoTIFFs from all subsets are mosaicked and tiled into 512×512 images at a ground sampling distance of 0.619 m/pixel. Each tile covers approximately 317×317 m, and adjacent tiles overlap by 50% (a stride of 256 pixels). The resulting gallery contains 15,130 georeferenced tiles. Mosaicking and latitude-dependent projection make the center grid not perfectly regular: all 15,130 centers are unique, but nearest-neighbor separations range from 2.45 m (mosaic-boundary duplicates of the same area) to a median of 153.7 m, and about 7% of centers have a neighbor within 79 m. The transition-cost correction of Section 3.4 is defined against this measured geometry rather than an idealized stride. Tiles corresponding to the training, validation, and test regions remain in a single gallery. All 1215 test queries search this full gallery, with tiles outside the test regions serving as distractors.

4.2. Evaluation Metrics and Cold-Start Protocol

C@K is the proportion of queries for which the top-K results contain at least one strong positive as defined by Equation (9); we report C@1. Candidate coverage is reported as R@50 (also called top-50 recall): the proportion of frames whose top-50 retrieval list contains at least one strong positive. We use the term R@50 consistently for this quantity wherever it appears. Localization error is the haversine distance between the center of the output tile and the ground-truth UAV position. We report its median and the proportion of queries with an error below 300 m ($D < 300$ m). All framewise metrics are computed from the causal output in Equation (3) and pooled over the 1215 queries from both test sequences; because the two sequences are imbalanced (1071 and 144 frames), we additionally report per-sequence results wherever pooled numbers are given, and the pooled values should be read as dominated by subset 02. The metric cutoff K and the candidate count K_{ret} denote, respectively, the evaluation rank and the size of the online candidate set.

First stable fix measures the time required to obtain a sustained valid position from an empty localization state. Each frame in a test sequence is treated in turn as a new start, with the position belief cleared at that frame. VO remains an independent causal observation stream, so its trailing filter state is not reset with the position belief. Let $d_{\text{geo}}(\cdot, \cdot)$ denote the geographic distance above and ϵ an error threshold. Starting from frame r, the first time at which L consecutive estimates satisfy the threshold is

$$T_{\mathrm{fix}}(r;\epsilon,L)=\min_{i\geq r+L-1}\{t_i-t_r\mid d_{\mathrm{geo}}(\hat{p}_\ell,p^\star_\ell)\leq\epsilon,\ \ell=i-L+1,\ldots,i\}. \tag{23}$$

We use $L=3$, $\epsilon\in\{50,100,300\}$ m, and a maximum observation period of 600 s. A start that does not achieve a stable fix within the available interval is censored at $\min(600\,\mathrm{s}, t_{\mathrm{end}}-t_r)$, where t_{end} is the end of its test sequence. Success rates use all 1215 starts as the denominator. Kaplan–Meier curves [33] summarize the first-stable-fix process while retaining censored starts; the restricted mean waiting time truncated at 600 s (RMST_{600}) integrates the Kaplan–Meier curve to the horizon [34], and uncertainty is quantified with the circular moving-block bootstrap [35,36] over time-ordered starts (block length 32; 5000 replicates for the framewise comparisons of Section 5.1 and 2000 replicates for the start-level comparisons of Section 5.3). Under the L-consecutive rule, an isolated single-frame outlier cannot by itself terminate or create a stable fix: an outlier inside a run merely delays it, and a single threshold violation after two consecutive hits resets the run. Section 5.3 additionally reports the sensitivity of the cold-start conclusions to $L\in\{1,5\}$ and to a relaxed "4-of-5" rule, together with the per-frame probability that an established fix is subsequently lost.

4.3. Model Training and Online Localization Configuration

The default cross-view retriever uses DINOv3 ViT-B/16 initialized with the LVD-1689M pretrained weights [5]. UAV and satellite images are resized to 512×512 and encoded by the shared backbone and descriptor head into 256-dimensional representations. VC-LoRA is inserted into the final six Transformer blocks with ranks $\rho\in\{4,4,8,8,12,12\}$ from shallow to deep and scaling factor $\alpha_{\mathrm{lora}}=16$, adding 1.88 million trainable parameters. Except in the cross-frontend experiment, this retriever supplies the candidates used by TRACE-Loc.

The retriever uses a frozen DINOv3 ViT-B/16 backbone (LVD-1689M weights) with 512×512 inputs. Trainable capacity consists of convolutional LoRA adapters on the last six transformer blocks—per-block ranks 4, 4, 8, 8, 12, and 12 with $\alpha=16$ and zero dropout, applied to attn.qkv, attn.proj, mlp.fc1, and mlp.fc2—where the fc1 adapters are shared between the two views and the fc2 adapters are view-specific (shared-residual strategy); the descriptor head is shared across views. The retriever is trained for 60 epochs with AdamW in bf16 mixed precision: learning rates are 10^{-4} for the descriptor head and 5×10^{-5} for the LoRA adapters, weight decay 0.05, three warmup epochs followed by cosine decay, per-GPU batch 20 on two GPUs (global batch 40), and training seeds $\{0,1,2\}$ with hash randomization pinned. The descriptor head pools the patch features with learnable GeM pooling ($p=3$) and spatial-attention pooling alongside the projected CLS token, fuses the three with weights $1/4$, $1/4$, and $1/2$, and applies one shared bias-free linear map to 256 dimensions followed by L_2 normalization. Training minimizes the one-way UAV$\rightarrow$SAT multi-positive InfoNCE objective described above (temperature 0.07, hard-negative weight 1.0). Augmentations are mild and asymmetric: the UAV view receives color jitter (brightness/contrast/saturation 0.15, hue 0.03), grayscale $p=0.05$, affine perturbations ($\pm8^\circ$, 5% translation, scale 0.95–1.05), and random erasing ($p=0.15$, area 0.02–0.06); the satellite view receives stronger color jitter (0.3, hue 0.1), grayscale ($p=0.1$), Gaussian blur ($p=0.3$, $\sigma\in[0.5,2]$), JPEG compression ($p=0.3$, quality 50–95), and a vegetation color shift ($p=0.4$); neither view is horizontally flipped. Three independent training runs (seeds 0, 1, 2) estimate frontend variation.

Online localization receives images, headings, timestamps, and relative altitude in chronological order. UAV-VisLoc does not provide raw barometer or range-sensor measurements. We therefore construct a relative-altitude proxy during offline preprocessing by subtracting the DEM elevation at the logged position from the logged altitude. Frame-to-frame speed is not estimated when the DEM sample is invalid or outside the raster.

At runtime, the localization algorithm reads only this scalar proxy, not the logged coordinates or the DEM. Logged coordinates are otherwise used to construct training labels and calculate localization errors.

Visual speed is estimated from consecutive images before heading alignment. Camera intrinsics are obtained from DJI Terra calibration. As reusable distortion coefficients are unavailable, no additional undistortion is applied. Test subsets 02 and 10 have original image sizes of 3976×2652 and 3000×2000, respectively, with intrinsic matrices

$$\mathbf{K}_{02} = \begin{bmatrix} 4037.294 & 0 & 2013.107 \\ 0 & 4037.294 & 1322.641 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{K}_{10} = \begin{bmatrix} 2987.884 & 0 & 1469.703 \\ 0 & 2987.884 & 1004.114 \\ 0 & 0 & 1 \end{bmatrix}. \tag{24}$$

After the center crop and resize in Equation (14), the effective focal lengths are 779.447 pixels for subset 02 and 764.898 pixels for subset 10.

Frame-to-frame motion is estimated from ORB matches using a RANSAC partial-affine model. Up to 1800 features are extracted from each image pair, and the 1200 matches with the smallest Hamming distances are retained; the RANSAC reprojection threshold is 5 pixels. An estimate is accepted when it has at least 80 matches, 40 inliers, and an inlier ratio of 0.15. The constant-speed prior $v_{\text{fb}} = 20\,\text{m/s}$ is used when geometric estimation fails, relative altitude is invalid, or the estimated speed falls outside $(0, 45]\,\text{m/s}$. When the frame interval exceeds 10 s, the online state is reinitialized from the current retrieval candidates.

The main experiments use $H = 3$, retaining the three most recent historical state sets. Each frame contributes the top-50 retrieval candidates, while the ten lowest-cost states from each historical set are projected, adding at most 30 candidates before merging and deduplication. No more than 80 states are retained after the current and projected candidates are combined. Heading and speed weights are 3.0 and 0.2. The speed-deviation scale σ_v is validation-selected like every other hyperparameter: a grid over $\{4, \ldots, 12\}\,\text{m/s}$ selects $\sigma_v = 8.0\,\text{m/s}$ as the interior optimum (validation C@1 77.93%, against 77.83% at both 7 and 10 m/s and 77.54% at 12 m/s), which is used for all reported results; Table 6 reports the full $\sigma_v \in \{2, 4, 6, 8, 12\}\,\text{m/s}$ sensitivity, including the earlier per-subset heuristic as its marked row. An adaptive rule based on the trailing MAD of the speed stream was also evaluated and underperformed (53.83% vs. 61.23%): the within-flight speed stream is too steady for its local dispersion to reflect the retrieval-to-retrieval speed mismatch that σ_v must express. The transition coefficient, upper bound on the projected score, and projection decay are $\lambda_T = 0.008$, $\eta = 0.4$, and $\tau_{\text{proj}} = 0.25$, respectively. Score calibration ties these constants to measurable distributions: over the test queries, the top-1 cosine similarity has mean 0.734 (median 0.743) and the rank-50 candidates average 0.508 (25th percentile 0.479), so the projection bound $\eta = 0.4$ sits near the middle of the top-50 score range—above the weakest candidates it must not outrank, below the current top-1 it must not displace—and the small λ_T compensates for the squared scale of the speed term in Equation (19), whose numerator is a distance residual that can reach tens of meters while E_{node} is bounded by one. This configuration is used for the main results, independent training runs, first-stable-fix evaluation, and input-perturbation experiments.

The history length, transition weight, upper bound on the projected score, and decay coefficient are selected on the validation set. The selected values, $H = 3$, $\lambda_T = 0.008$, $\eta = 0.4$, and $\tau_{\text{proj}} = 0.25$, are fixed before testing. The same grid was re-run under the corrected transition cost of Section 3.4; we retained the originally selected configuration for all revision experiments (so that every reported number corresponds to one frozen method definition), and we report the full re-selection grid, whose validation argmax differs (validation C@1 78.32% vs. 77.93%), as a sensitivity analysis in Appendix D together with its test behavior.

Training and inference use two and one NVIDIA RTX 4090 D GPUs, respectively. CPU timings are measured on an Intel Xeon Gold 6346. PyTorch, FAISS, and OpenCV are used for model inference, gallery retrieval, and visual-speed estimation (Python 3.11, PyTorch 2.10.0 with CUDA 12.8, FAISS 1.13.2, OpenCV 4.13.0, NumPy 2.4.3, SciPy 1.17.1).

4.4. Comparison Methods and Statistical Analysis

Temporal comparisons include causal score averaging (seq-avg), the belief recursion without historical projection (beam Viterbi; Section 3.4), a 256-particle filter, a forward constant-velocity Kalman filter (CV-KF), and belief variants using either fixed speed or visual speed. All methods receive the same top-50 candidates in chronological order, with their parameters selected on the validation set; Appendix A specifies every state model, observation model, noise setting, initialization, resampling rule, and output selection.

The cross-frontend experiment changes only the candidate generator. To remove differences in descriptor-score scale, ranks 1–100 from each frontend are mapped within each query to scores from 0.75 to 0.45, and the upper bound on the projected score is set to $\eta = 0.3$. The comparison includes Sample4Geo-ZS with its University-1652 weights, Sample4Geo-FT fine-tuned on our training split, and DINOv3 ViT-L/16 initialized with SAT-493M weights. All other experiments use cosine similarity from the default frontend.

Main results pool all 1215 queries from the two test sequences, with per-sequence results reported alongside wherever pooled numbers appear. Means and standard deviations are calculated from three independent training runs. Paired differences and their 95% confidence intervals use the outputs of the first run and 5000 circular moving-block bootstrap samples with a block length of 32; the same bootstrap scheme is applied at the start level (block length 32) for cold-start waiting-time comparisons. Temporal comparisons, input perturbations, and timing experiments use candidates generated by the same model. Appendix A documents the full configuration of every temporal baseline (state model, observation model, noise settings, initialization, resampling, and output selection), the frame-interval distribution of the test sequences, sequence durations, and the reset counts incurred by the 10 s rule; Appendix B lists all symbols, and Appendix C lists the exact split membership.

5. Results

Candidate-level trajectory constraints are intended to maintain a consistent position sequence when the quality of single-frame candidates fluctuates. We first evaluate their effect on continuous localization, then examine whether the same behavior persists across retrieval frontends and from an empty initial state. Candidate coverage, input conditions, and computational cost are analyzed to identify where the method is effective and where its limitations arise.

5.1. Overall Effect of Candidate-Level Trajectory Reranking

Table 1 compares temporal averaging, the no-projection belief recursion (beam Viterbi), a particle filter, forward CV-KF, and TRACE-Loc belief on the same top-50 candidates produced by the default VC-LoRA frontend. Every method receives identical candidates and motion observations in chronological order, and its parameters are selected on the validation set.

Table 1. Causal localization on a common set of global-retrieval candidates. ↑: higher is better; ↓: lower is better.

Method	C@1 (%) ↑	Median Error (m) ↓	D < 300 m (%) ↑	Time (ms) ↓
Single frame	49.05	150.92	53.66	–
Causal score averaging ($W = 9$)	16.05	611.00	27.74	0.198
Belief w/o projection (beam Viterbi) + VO	49.30	143.36	54.40	0.174
Particle filter + VO	58.11	96.38	63.54	0.483
Forward CV-KF + VO	51.85	118.85	56.63	**0.051**
Belief, fixed 20 m/s	59.67	94.16	68.15	1.902
TRACE-Loc + VO	**61.23**	**93.75**	**70.62**	1.913

Note. Time is CPU post-processing per frame over cached candidates and excludes candidate retrieval. A dash indicates that no temporal post-processing is used. Boldface marks the best applicable value in each column. "Belief w/o projection" is the recursion of Equations (16)–(22) with historical projection disabled ($K_{\text{nb}} = 0$); it coincides with a beam-width-limited Viterbi decode over the candidate trellis, and we use this label consistently throughout the revision. Both belief-based methods use the corrected transition cost of Equation (17); the particle filter and forward CV-KF use their own propagation models (Appendix A).

With visual speed, belief raises C@1 from 49.05% to 61.23%, reduces median error from 150.92 to 93.75 m, and improves D < 300 m by 16.96 percentage points. The particle filter has a 2.63 m higher median error, whereas belief is 3.12 and 7.08 points higher in C@1 and D < 300 m, respectively. Averaging similarity scores over time performs substantially worse because candidates at the same rank can correspond to different places in successive frames. Historical projection accounts for most of the gain over pure temporal selection: disabling it while keeping the same transition costs lowers C@1 to 49.30% and D < 300 m to 54.40% (Section 5.4).

Across three independent training runs, mean C@1 is 48.67 ± 0.41% for single-frame retrieval and 58.33 ± 2.84% for TRACE-Loc, a paired gain of 9.66 ± 2.69 percentage points; per-run paired gains are +12.18, +6.83, and +9.96 points, so the first run reported below is the most favorable of the three. All detailed comparisons in this paper use this first training run, so that no downstream experiment selects among runs. The projection stage itself contributes +11.93, +4.44, and +7.82 points over the identical recursion without projection on the same three runs (+8.07 ± 3.75 points overall). For the first run, the paired moving-block bootstrap gives a C@1 difference of +12.18 points (95% CI [8.23, 16.30]) and a median-error difference of −57.17 m (95% CI [−660.03, −15.22] m).

The pooled result combines two test sequences with different single-frame retrieval difficulty. On the 1071 frames of subset 02, trajectory constraints raise C@1 from 51.35% to 62.28% and reduce median error from 119.80 to 92.35 m; on subset 10, C@1 increases from 31.94% to 53.47%, while median error falls from 483.26 to 97.92 m. Both sequences improve, with the larger gain occurring on subset 10, where the single-frame baseline is weaker; the sequence-wise macro average of C@1 rises from 41.65% to 57.88%. Section 5.4 relates this difference to candidate coverage.

Figure 5 shows nine consecutive online localization frames from test subset 02, regenerated under the corrected transition cost and the frozen configuration; all numeric labels are localization errors (m or km as shown). The single-frame top-1 result is 411.1 and 460.8 km from the ground truth at frames $i - 6$ and $i - 5$. After historical states introduce nearby projected candidates, the online errors at these frames fall to 41.4 and 37.5 m. In another frame, the trajectory constraint reduces a 221.4 m single-frame error to 98.2 m. The sequence illustrates how candidate recovery operates during brief failures of the retrieval ranking.



Figure 5. Nine-frame reranking example from test subset 02. Rows show UAV images, retrieval candidates, projected candidates, and causal outputs; c1, c2, … denote current-frame retrieval candidates ordered by similarity, p1, p2, … denote candidates added by historical projection, and "Recovered" marks a frame whose online output returns within 300 m of the ground truth after an incorrect single-frame top-1.

5.2. Reranking Across Retrieval Frontends

Table 2 fixes the reranker and changes only the candidate-retrieval frontend. Query-wise rank mapping removes differences in descriptor-score scale, and all frontends use the same reranking parameters. The six reruns use the corrected transition cost, the causal visual-speed stream, one shared recursion configuration, and the frozen $\sigma_v = 8.0\,\mathrm{m/s}$; the VC-LoRA (LVD-1689M) row is evaluated on the canonical seed-0 cache, whose uncalibrated single-frame C@1 (49.05%) matches Table 1.

Table 2. Effect of causal reranking across retrieval frontends.

Frontend	C@1 (%)			Gain (pp)
	Single Frame	Fixed Speed	Visual Speed	
MixVPR [2]	8.48	19.75	24.53	+16.05
Sample4Geo-ZS [4]	10.62	11.85	13.33	+2.72
SALAD [3]	33.91	54.24	56.79	**+22.88**
Sample4Geo-FT [4]	8.23	16.46	16.95	+8.72
VC-LoRA (LVD-1689M)	**49.05**	**56.79**	**59.01**	+9.96
VC-LoRA (SAT-493M)	43.79	53.83	55.39	+11.60

Note. Gain is the absolute increase from single-frame C@1 to reranking with visual speed. Boldface marks the best value in each column.

VO belief improves C@1 for all six frontends, with gains ranging from 2.72 to 22.88 percentage points. This consistent improvement indicates that candidate-level trajectory constraints are not tied to one descriptor, although the absolute performance remains limited by the quality of the retrieved candidates. Reranking improves frontends initialized from either natural-image or satellite-image pretraining. The comparison here concerns the change before and after reranking, rather than the absolute difference between the two pretrained models. Because this table uses query-wise rank scores, the default frontend result is not directly comparable with Table 1, which uses its original cosine similarity.

5.3. First Stable Fix Under Cold Start

Under the per-start cold-start protocol in Section 4, single-frame retrieval and belief use the same retrieval frontend and the same 1215 starting points. Visual speed continues to use the trailing median from the current and preceding frames. Table 3 summarizes the probability of an early stable fix and the restricted mean waiting time at three distance thresholds.

Table 3. Cold-start first-stable-fix performance over 1215 starting points at three distance thresholds. ↑: higher is better; ↓: lower is better.

Method	50 m		100 m		300 m	
	P_{30} (%) ↑	RMST_{600} (s) ↓	P_{30} (%) ↑	RMST_{600} (s) ↓	P_{30} (%) ↑	RMST_{600} (s) ↓
Single frame	6.73	404.5	48.09	61.7	57.26	49.0
Belief w/o projection + VO	8.72	401.2	55.70	61.4	62.06	52.9
Forward CV-KF + VO	8.06	410.4	51.46	52.7	59.91	45.2
Particle filter + VO	**9.39**	**345.7**	56.97	**48.4**	**63.34**	**42.9**
TRACE-Loc + VO	9.05	384.3	**57.36**	105.5	63.15	95.3

Note. P_{30} is the Kaplan–Meier probability of a stable fix within 30 s. RMST_{600} is the restricted mean waiting time up to 600 s. Boldface marks the best value at each threshold. The no-projection row is the beam-Viterbi recursion of Table 1. Per-threshold censoring counts and per-sequence results are reported in Appendix A.

Relative to single-frame retrieval, TRACE-Loc increases P_{30} by 2.33, 9.27, and 5.89 percentage points at the 50, 100, and 300 m thresholds (paired moving-block bootstrap 95% CIs [+0.58, +4.52], [+5.02, +13.89], and [+2.76, +9.47]; all intervals exclude zero). The corresponding RMST_{600} differences are -20.2, $+43.8$, and $+46.3$ s. Per-threshold censoring counts and per-sequence results are reported in Appendix A; per-sequence 30-s fix probabilities for TRACE-Loc at 100 m are 58.1% (subset 02) and 51.3% (subset 10). The advantage is concentrated early in the cold start. At 100 and 300 m, TRACE-Loc has RMST_{600} values 43.8 and 46.3 s longer than single-frame retrieval, and the curves cross later in the observation period. At 50 m, the final proportion of starts achieving a stable fix rises from 48.72% to 56.87%, while the particle filter reaches 63.95%.

Relative to the particle filter, the two methods are statistically indistinguishable in P_{30} at every threshold, while the particle filter recovers faster over longer horizons (RMST_{600} differences $+38.6$, $+57.1$, and $+52.4$ s in its favor, significant at 100 and 300 m; the 50 m interval includes zero). Historical projection contributes to early acquisition relative to the no-projection ablation (P_{30} differences $+0.33$ [-0.92, $+1.67$], $+1.66$ [$+0.17$, $+3.46$], and $+1.08$ [-0.00, $+2.22$] points at 50, 100, and 300 m; only the 100 m interval excludes zero), but also prolongs recovery when an incorrect path persists. The claims of this paper are therefore limited to early acquisition over single-frame retrieval; they do not establish an advantage over the particle filter at any horizon.

The stability criterion itself does not drive the conclusions. Replaying the identical cold-start protocol under alternative criteria, the 30-s fix probability at 100 m moves from 57.36% at the published $L = 3$ to 67.26% at $L = 1$, 35.13% at $L = 5$, and 51.48% under a relaxed "4-of-5" rule (at 300 m: 63.15%, 68.86%, 56.05%, and 59.12%); single-frame retrieval rescales by similar amounts (100 m: from 48.09% at $L = 3$ to 70.26%, 26.23%, and 40.29%), so the method ordering is preserved while the absolute fractions rescale as expected. The same replay yields a direct stability statistic: among frames where the current estimate is already within the threshold, the probability that the next frame leaves it is 17.5% for TRACE-Loc versus 24.3% for single-frame retrieval at 100 m, and 3.9% versus 13.5% at 300 m: the recursion outputs are markedly more stable once a fix is established.

5.4. Sources of the Trajectory-Reranking Gain

For online trajectory constraints to help, the current candidate set must first retain tiles related to the true position. Table 4 compares single-frame retrieval and TRACE-Loc by test region; the subsequent analysis relates these regional outcomes to candidate coverage.

Table 4. Regional comparison of single-frame retrieval and TRACE-Loc.

Region	Frames	C@1 (%)		Gain (pp)	Median Error (m)
		Single Frame	TRACE-Loc		
All test data	1215	49.05	61.23	+12.18	93.75
Subset 02, land	734	73.71	86.38	+12.67	65.84
Subset 02, open water	337	2.67	9.79	+7.12	631.85
Subset 10	144	31.94	53.47	+21.53	97.92

Note. Gain is the absolute C@1 improvement over single-frame retrieval; median error is reported for TRACE-Loc.

The regional results reveal three different regimes. The land segment of subset 02 has 96.19% top-50 coverage, and trajectory constraints raise C@1 to 86.38%. Subset 10 also has high top-50 coverage but only 31.94% single-frame C@1; reranking raises it to 53.47%, indicating that unstable candidate ordering is the main issue in this region. Over open water, top-50 coverage is only 13.95%, online C@1 is 9.79%, and median error remains 631.85 m. The limiting factor there is candidate generation, and trajectory constraints can help only on frames that retain a useful candidate. Because pooled metrics are dominated by subset 02, we also report the sequence-wise macro average of C@1, which rises from 41.65% to 57.88% under the same correction.

Increasing K_{ret} from 50 to 100 raises overall candidate coverage from 72.35% to 76.54% but reduces online C@1 to 45.35%. Additional candidates are subject to the $B_{\max} = 80$ truncation and include more weak matches, so a wider candidate set does not necessarily provide better candidates for online selection. Because this configuration changes two factors at once, we decouple them with a $K_{\mathrm{ret}} \times B_{\max}$ grid at the frozen configuration (Table 5): at $K_{\mathrm{ret}} = 50$, raising $B_{\max}$ from 80 to 160 changes nothing, while at $K_{\mathrm{ret}} = 100$ the loss is mostly truncation—raising $B_{\max}$ to 120 already recovers C@1 from 45.35% to 59.01%—with the remaining 2.2-point gap to $K_{\mathrm{ret}} = 50$ attributable to the weaker extra candidates themselves. Candidate count and state capacity are therefore separable effects, and neither redundancy nor capacity alone explains the preference for $K_{\mathrm{ret}} = 50$.

Table 5. Decoupling retrieval candidate count K_{ret} from belief capacity $B_{\max}$ (C@1 %, median error m, D < 300 m %).

	$B_{\max} = 80$	$B_{\max} \in \{120, 150, 160\}$
$K_{\mathrm{ret}} = 50$	61.23/93.75/70.62	identical to $B_{\max} = 80$
$K_{\mathrm{ret}} = 100$	45.35/265.30/50.53	59.01/96.76/67.41

Candidate coverage defines the search range supplied by the current retrieval, whereas historical projection determines whether earlier states can extend that range at the current frame. To isolate this mechanism, we keep the current candidates and transition cost unchanged and compare $K_{\mathrm{nb}} = 0$ with $K_{\mathrm{nb}} = 10$. Without historical projection, C@1 and D < 300 m are 49.30% and 54.40% (the same frozen configuration with $K_{\mathrm{nb}} = 0$; this is also the Table 1 baseline row); projecting ten states raises them to 61.23% and 70.62%. In the framewise comparison, projection corrects 149 C@1 errors relative to $K_{\mathrm{nb}} = 0$, while changing 4 correct outputs to incorrect ones, for a net gain of 145 correct outputs.

The gain from projection does not come solely from restoring a correct tile omitted by current retrieval. Among the 1215 test frames, 336 have no strong positive in the current

top-50 list. Projection directly restores C@1 on 28 of these frames and D $<$ 300 m on 64. The remaining gain arises from the effect of projected candidates on subsequent path selection. Conversely, an incorrect projection can occupy limited state capacity and delay recovery, consistent with the long cold-start tail of TRACE-Loc in Figure 6.

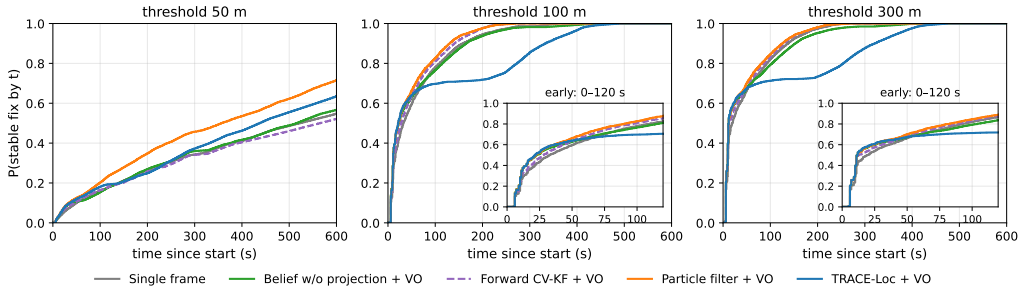


Figure 6. Kaplan–Meier first-stable-fix curves for five causal localization methods under cold start. Curves report the fraction of starts localized within each distance threshold; incomplete starts are right-censored.

Two further analyses separate grid discretization from path-selection behavior. First, the discrete output space itself imposes an error floor: even an oracle that always reports the tile center nearest to the ground-truth position achieves a median error of 48.27 m (mean 47.39 m, P90 68.82 m, maximum 89.05 m), and every test frame lies within 100 m of some tile center. A coarser stationary guard that also freezes the heading term below half the median grid step (76.9 m) reproduces every main metric exactly at the frozen configuration. TRACE-Loc's 93.75 m median therefore contains a quantization component of roughly half its magnitude, and residual errors beyond the floor measure tile mis-selection rather than representation quality. Second, the cumulative cost in Equation (20) can suppress alternative hypotheses during difficult recovery: the discounted variant of Equation (21) trades early persistence against late flexibility: every discount degrades framewise accuracy substantially and non-monotonically (C@1 53.99% at $\gamma = 0.8$, 51.60% at 0.85, 48.31% at 0.90, and 55.31% at 0.95, against 61.23% at $\gamma = 1$), so the frozen $\gamma = 1$ remains preferable at this operating point. A beam-composition diagnostic shows the same trend: flattening accumulated-cost differences raises the mean projected-candidate score from 0.206 at $\gamma = 1$ to 0.234 at 0.90 and the projection-origin share of the retained state set from 16.9% to 17.5%, with the largest share at $\gamma = 0.90$. This association is consistent with, but does not establish, the proposed explanation.

Table 6 examines the effects of candidate count, history length, and transition weight. Each experiment changes one parameter while holding the remaining configuration fixed. The validation-selected configuration used for the main results is identified in the table.

C@1 rises as the candidate count increases from 1 to 50. At 100 candidates, weak additions and truncation at $B_{\max} = 80$ jointly disrupt path selection (Table 5). A larger H allows more earlier states to enter the current path comparison: increasing H from 3 to 12 improves test C@1 by 0.66 points, while requiring more projections and a longer retained history; the main results continue to use $H = 3$, as selected on the validation set. Once the transition weight exceeds 0.008, the current visual observation has progressively less influence on accumulated cost and all three metrics decline. The projection score η shifts test C@1 across its grid $\{0, 0.2, 0.3, 0.4, 0.5\}$ by at most 2.2 points and its decay τ_{proj} by at most 0.7 points. Setting the projected-score bound to $\eta = 0$—projected candidates still extend the candidate set, but without score priority—retains 60.08%, so most of the projection benefit comes from extending the candidate set rather than from the projected score itself. Together, these trends show that the candidate count sets the available search

range, while H and the transition weight control the influence of historical states on the current output.

Table 6. Sensitivity to candidate count, history length, transition weight, no-projection history, and speed-deviation scale. ↑: higher is better; ↓: lower is better.

Parameter	Value	C@1 (%) ↑	Median Error (m) ↓	D < 300 m (%) ↑
K_{ret}	1	48.81	149.7	59.09
	5	55.80	105.6	61.98
	10	57.04	102.6	63.54
	20	60.08	94.8	67.08
	50	61.23	93.8	70.62
	100	45.35	265.3	50.53
H	1	59.84	96.5	68.72
	3	61.23	93.8	70.62
	5	61.23	93.8	70.62
	8	61.65	92.2	71.19
	12	61.89	92.1	71.44
λ_T	0.004	57.53	101.4	66.34
	0.008	61.23	93.8	70.62
	0.012	60.25	94.0	70.12
	0.020	59.26	96.4	68.40
	0.035	55.88	104.7	65.43
K_{nb}	0	49.30	143.4	54.40
	5	60.58	94.3	69.88
	10	61.23	93.8	70.62
	20	61.32	93.6	70.78
σ_v (m/s)	2	47.33	153.3	58.02
	4	56.46	105.5	66.50
	6	60.25	94.2	69.88
	8	61.23	93.8	70.62
	12	56.71	103.4	65.51
	6.0/4.0 †	60.16	94.3	70.12

Note. Bold parameter values are selected on the validation set and used in the main experiments; † marks the submitted subset-specific pair (6.0 on subset 02, 4.0 on subset 10). All reported metrics are evaluated on the test set. $B_{\max}$, η, and τ_{proj} grids are reported in the text and in the released diagnostics.

5.5. Effect of Motion-Observation Errors

Table 7 evaluates how errors in heading, relative altitude, and visual speed affect trajectory propagation. The baseline uses the default inputs without artificial perturbation. Heading noise is sampled independently at each frame, relative-height bias is fixed within a sequence, and VO dropout replaces the stated fraction of valid speed observations with the constant-speed prior from Section 4.

At a heading-noise standard deviation of 30°, C@1 decreases by 2.30 percentage points. Relative-height biases within ±50 m shift C@1 by between −0.74 and +0.17 points. Complete VO dropout reduces C@1 and D < 300 m by 1.56 and 2.47 points, whereas speed noise up to 10 m/s changes C@1 by at most 1.23 points. For trajectory propagation, projection direction and the availability of a speed observation matter more than modest speed fluctuations, while the constant-speed fallback limits the effect of short VO outages.

The heading-noise rows above perturb only trajectory propagation—the projection direction and the transition cost—while image pre-alignment (Equation (6)) still uses the logged heading. When the same noise realization is additionally applied to image pre-alignment, C@1 falls to 59.09% at 10° (a further −1.24 points relative to propagation-

only) and to 51.77% at 20° (a further −8.07 points, with median error rising from 94.8 to 119.2 m). Heading noise therefore has a modest effect on propagation but a large effect once it corrupts the north-up pre-rotation; robustness claims in an operational system must account for this second pathway, which an AHRS-grade heading source keeps small. The tested sequences also contain little aggressive maneuvering: the per-frame body turn rate $|\Delta\psi|/\Delta t$ has median 0.56°/s, 90th percentile 1.61°/s, and maximum 6.15°/s. Binning the framewise outputs by turn rate gives C@1 of 61.5% below 2°/s (1146 frames) and 57.4% in 2–5°/s (68 frames). We additionally evaluated a turn-triggered history reset for such regimes: clearing all historical states whenever an inter-frame heading change exceeds $\{5, 10, 15\}^\circ$ *per interval* reduces C@1 to 53.6%, 48.7%, and 58.6% respectively. The non-monotonicity across thresholds is expected: at 10° per interval the reset fires often enough to discard correct hypotheses (each trigger restarts localization from the current top-50 list), whereas at 15° it still costs 2.6 points relative to the no-reset run (58.6% vs. 61.23%). Above 20° per interval the sweep becomes bit-identical to the no-reset run: only 17 of the 1213 test intervals (1.4%) reach $|\Delta\psi| \geq 20^\circ$, and 16 of them are the long (>10 s) observation gaps where the 10 s rule has already cleared the temporal state, so the additional reset is a no-op. On these sequences the retained $H = 3$ history is already conservative relative to the maneuver envelope, and discarding it discards correct hypotheses more often than incorrect ones; an adaptive reset may still be preferable in platforms with genuinely aggressive turn profiles, but our data do not demonstrate a benefit.

Table 7. Sensitivity to heading noise, relative-height bias, VO dropout, and speed noise. ↑: higher is better; ↓: lower is better.

Perturbation	Magnitude	C@1 (%) ↑	Median Error (m) ↓	D < 300 m (%) ↑
Baseline	None	61.23	93.8	70.62
Heading-noise standard deviation	5°	61.07	93.8	70.70
	10°	60.33	94.5	69.38
	20°	59.84	94.8	68.72
	30°	58.93	97.1	67.82
Relative-height bias	−50 m	60.49	93.8	69.79
	−25 m	60.82	93.8	70.21
	+25 m	61.40	92.7	70.95
	+50 m	61.15	92.7	70.86
VO dropout	25%	60.41	94.2	69.38
	50%	60.16	93.9	69.14
	75%	59.67	94.3	68.48
	100%	59.67	94.2	68.15
Speed-noise standard deviation	2 m/s	60.41	94.0	69.55
	5 m/s	60.00	93.8	69.38
	10 m/s	60.49	92.7	69.71

Note. The baseline uses unperturbed inputs. Heading noise is sampled independently at each frame, relative-height bias is fixed within a sequence, and VO dropout replaces valid speed observations with the constant-speed prior.

5.6. Online Computational Cost

The times in Table 1 cover only CPU post-processing over cached candidates; this replay includes candidate merging, deduplication, projection, the full transition-cost evaluation, belief sorting, and output selection. TRACE-Loc belief takes 1.913 ms per update in this replay; the no-projection variant, the particle filter, and forward CV-KF take 0.174, 0.483, and 0.051 ms, respectively. All temporal methods remain below 2 ms. The complete online path—image reading and decoding, preprocessing, DINOv3 encoding, ORB/RANSAC visual speed, FAISS retrieval, and the belief update—has a mean latency of 187.17 ms,

corresponding to 5.34 frames/s. Within this total, image reading accounts for 38.5 ms, preprocessing (rotation, crop, resize) for 76.4 ms, DINOv3 encoding for 17.4 ms, VO for 51.7 ms (27.6%), FAISS for 1.3 ms, and the belief update for 1.9 ms (1.02%), measured in-pipeline at the frozen configuration; the replay figure (1.913 ms) and the in-pipeline belief update (1.9 ms) agree because both now refer to the same frozen configuration, and the earlier discrepancy between a replay figure and a benchmarked share arose from an end-to-end benchmark that had been configured with a different, non-frozen parameterization. Throughput is governed mainly by image decoding, preprocessing, and VO rather than by trajectory updating.

Because framewise VO is the largest online component, Table 8 examines the effect of reducing its update frequency. Here, N is the interval between VO updates; intermediate frames reuse the most recent valid speed and fall back to 20 m/s until a valid observation becomes available. All belief parameters remain unchanged.

Table 8. Localization accuracy under sparse VO updates (corrected transition cost, frozen configuration). ↑: higher is better; ↓: lower is better.

N	Localization		Median Error (m)	Scheduled VO	Amortized VO
	C@1 % ↑	D < 300 m % ↑		Calls	Time (ms) ↓
1	61.23	70.62	93.8	1213	51.7
2	61.48	70.70	92.4	607	25.9
3	61.40	70.62	92.4	405	17.2
5	62.30	71.69	92.1	243	10.3

Note. Scheduled calls count the ORB/RANSAC invocations under the N-interval schedule; amortized VO time is the measured framewise VO cost (51.7 ms) scaled by the schedule. All main comparisons use $N = 1$; end-to-end latency at $N = 1$ is 187.17 ms (Section 5.6), of which the remaining stages are independent of N.

Updating VO every second frame reduces the number of scheduled ORB/RANSAC calls from 1213 to 607, every third frame to 405, and every fifth frame to 243. Under the corrected transition cost, C@1 changes little across schedules (61.23% at $N = 1$, 61.48% at $N = 2$, 61.40% at $N = 3$, and 62.30% at $N = 5$, with median error between 92.1 and 93.8 m), so on these sequences the visual-speed schedule is not a sensitive factor and the holding-plus-fallback rule compensates reliably for skipped updates. All main results use $N = 1$ so that scheduling does not affect the method comparisons; a deployment seeking to reduce the largest online cost can adopt $N = 2$ or $N = 3$ at essentially unchanged accuracy. The efficiency gain studied in this work is earlier acquisition of a stable absolute position from an empty state; framewise throughput remains dominated by the shared image-processing, encoding, and VO frontend.

5.7. Comparison with Trajectory-Level Alignment (NaviLoc)

NaviLoc [29] is the closest trajectory-level alternative: it aligns the complete relative trajectory to the reference map with a global SE(2) estimate, refines it with sliding-window bounded Procrustes, and smooths the result, reporting large gains on a non-public low-altitude benchmark (data available from the authors under NDA). Its code is likewise available only under NDA, so we re-implemented its three stages from the published algorithms with fixed hyperparameters ($K = 72$ angles, window $W = 10$, stride $S = 7$, $|\theta| \leq 0.09$ rad, outlier gate $z < -1.5$) and two documented adaptations beyond the substitutions below: its local search radius is set to $r = 158.5$ m (NaviLoc tunes $r = 150$ m to its 40 m map spacing; our value matches the half-tile spacing of our grid), and its DeiT–Tiny descriptors are replaced by our VC-LoRA descriptors so that both methods read the same candidate list. First, UAV-VisLoc provides no IMU, so the relative trajectory is integrated from the same causal visual speed and logged heading that TRACE-Loc uses. Second, our

gallery spans all regions rather than a single local mosaic, so each window's translation is initialized from the densest cluster of per-frame top-1 retrieval anchors before the published in-radius refinement.

These substitutions bound the scope of the comparison. Integrating visual speed over a full sequence accumulates odometric drift: the median discrepancy between the integrated relative trajectory and the ground-truth relative trajectory grows from 131 m (first 20 frames) to 2.3 km (frames 101–200) on subset 02, and reaches 6.4 km over the full 1071-frame sequence (2.0 km on the shorter subset 10). A rigid SE(2) alignment of the complete trajectory is therefore not applicable in our setting—not as a failure of the method, but because the required VIO-grade relative trajectory is unavailable—so we adopt the sliding-window operation NaviLoc itself recommends for long flights (keeping roughly the most recent 2–3 km of context) and align independent windows of N frames, evaluating every frame by its window's estimate; at our speed scale (about 20 m/s over 3–6 s intervals) $N = 10$ spans roughly 0.8 km and $N = 20$ roughly 1.6 km, so $N = 20$ already approaches the lower end of that guidance, and longer windows degrade because integrated-VO drift grows with window length. As a diagnostic only—it uses test labels and is not part of the method or of any reported comparison—we re-initialize each window's translation with the ground-truth-optimal anchor: this yields a 145.2 m median error at $N = 10$ on subset 02, far below the 773.6 m obtained from retrieval-anchored initialization, indicating that anchor quality under visual drift accounts for a substantial part of the gap, although other implementation differences cannot be excluded.

Under this adapted protocol, trajectory-level alignment does not outperform framewise retrieval on our data. On subset 02, single-frame top-1 achieves a 119.8 m median error, while window alignment yields 773.6 m at $N = 10$ and degrades with longer windows (1108.5 m at $N = 20$) as within-window drift grows. On subset 10—a compact 2.3 km trajectory closest to NaviLoc's operating regime—window alignment at $N = 10$ is competitive (402.3 m vs. 483.3 m single-frame median) but loses its advantage as N grows. For reference, TRACE-Loc attains 92.35 m (subset 02) and 97.92 m (subset 10) median error under the same causal protocol, so the online stage outperforms both framewise retrieval and the adapted window alignment on these data. Three scope differences remain and preclude a direct transfer of NaviLoc's reported 19.5 m accuracy—a mean error over its single 2.3 km, 58-frame evaluation trajectory, whereas our comparisons report medians: it uses a synchronized IMU, outputs continuous coordinates on a small single-region map, and optimizes the complete trajectory offline, whereas our protocol requires framewise causal outputs from a global multi-region gallery with a lower-grade visual-speed substitute. The two approaches are complementary rather than interchangeable: a trajectory-level stage of this kind presupposes exactly the stable global candidate and a VIO-grade relative trajectory that an online initial-localization stage such as TRACE-Loc is meant to provide first, and it can refine continuous coordinates once a stable candidate exists (Section 6).

6. Discussion

6.1. Role of Candidate Projection and Path Selection

The comparisons on identical candidate sets show that the gain of TRACE-Loc comes from spatial transitions between candidates rather than simple temporal accumulation of similarity. Temporal averaging mixes candidates that share the same rank but represent different places. The no-projection variant, forward CV-KF, the particle filter, and TRACE-Loc instead use explicit spatial relations between successive candidates; the no-projection variant is exactly a beam-width-limited Viterbi decode over the retrieval trellis, which locates our contribution in the historical projection stage rather than in temporal selection itself. TRACE-Loc retains multiple low-cost paths and projects recent states to the current

time. When the single-frame ranking fails briefly, location hypotheses consistent with the established direction of travel can still enter the current path comparison, as illustrated by the recovery sequence in Figure 5.

Historical projection does more than replace tiles omitted by the current retrieval. More often, projected candidates alter path competition at the current and subsequent frames, allowing a motion-consistent hypothesis to survive a short retrieval failure. Candidate generation and trajectory selection are therefore complementary: the former determines the set of locations available for evaluation, and the latter uses successive observations to choose among them. All six retrieval frontends benefit from reranking, indicating that this candidate-level constraint is not specific to one descriptor, although absolute performance remains dependent on candidate quality [2–4]. Incorrect projections can likewise occupy the finite beam and influence later comparisons: relative to the no-projection recursion they correct 149 framewise outputs while harming 4, and directly rescue only 28 of the 336 frames whose current retrieval misses every strong positive, accounting for the slower recovery of some difficult starts.

6.2. Early Localization and Later Recovery Under Cold Start

Table 3 and Figure 6 reveal two phases of cold-start behavior. Early in the sequence, short-term motion consistency helps establish a stable candidate path: relative to single-frame retrieval, the 30-s stable-fix probability is higher at every threshold, with paired bootstrap intervals excluding zero at every threshold. As observations accumulate, a small number of incorrect historical states continue to compete with newer paths, causing the RMST curves to favor the particle filter and, at long horizons, single-frame retrieval. First localization is therefore not fully characterized by a single success rate or by waiting times calculated only from successful starts. The early stable-fix probability measures whether a usable position appears promptly, while restricted mean waiting time with censoring captures recovery over the more difficult starts. Together, the two measures describe both early acquisition and the subsequent tail.

The later recovery difference between TRACE-Loc and the particle filter follows from their state representations. Particle filters propagate a continuous position distribution [11–13], whereas TRACE-Loc retains a finite set of paths between discrete tiles produced by global retrieval. The latter connects directly to a cross-view retriever and provides an output at every frame, but an incorrect path can remain in the beam. The particle filter maintains a more dispersed set of position hypotheses, consistent with its faster later recovery on difficult starts. We emphasize that our cold-start results establish an early-acquisition advantage over single-frame retrieval only; against the particle filter, early-fix probabilities are statistically indistinguishable and long-horizon waiting times are worse, so the particle filter remains the stronger choice when a continuous prior and longer observation windows are available. Unlike methods that jointly align complete reference and query sequences [9,10,29], each TRACE-Loc update uses only observations that have already arrived, addressing online initial localization rather than offline trajectory alignment (Section 5.7).

6.3. Operating Scope and Deployment Conditions

Trajectory projection uses heading, time interval, relative altitude, and visual speed. The perturbation results show that heading error and the availability of speed observations affect propagation more directly than the tested height biases, whereas heading noise that also corrupts the image pre-rotation degrades accuracy substantially at 20° (Section 5.5). In an operational system, these quantities can be supplied by an AHRS, a barometer or range sensor, and onboard VO. Equation (11) approximates motion direction over a

short interval. Sharper turns and lateral slip reduce the useful projection horizon; on the tested sequences a turn-triggered history reset never improved accuracy (Section 5.5), but platforms with genuinely aggressive maneuver profiles may require a shorter history or confidence-triggered reset. Sparse VO scheduling further shows that reducing the update frequency can lower the largest online cost, while candidate reranking itself is not the throughput bottleneck.

Height Input: Offline Proxy Versus Onboard Sensing

The relative altitude used for visual-speed estimation is constructed offline by subtracting the DEM elevation at the logged position from the logged altitude. This proxy is available only because the evaluation replays logged flights; a GNSS-denied deployment would obtain relative height from a barometer, a radar or laser rangefinder, or terrain-referenced altimetry. The reranker consumes only the resulting scalar $\hat{v}$ and never the logged coordinates, so the offline construction affects the *quality* of one input, not the information set of the method. Its error propagates linearly: a relative bias in h scales $\hat{v}_\ell$ (Equation (15)) by the same factor, and the projected distance $d_{q \to i}$ (Equation (11)) scales proportionally over the propagation interval. The perturbation experiments bound this channel empirically: fixed biases of ± 50 m shift C@1 by between -0.74 and $+0.17$ points, a drifting bias mimicking an uncalibrated barometer at 0.02–0.1 m/s (6–30 m after five minutes of flight) keeps C@1 within 1.3 points of the baseline, and replacing valid speed observations wholesale with the 20 m/s fallback (a stronger perturbation than the fixed biases tested here) costs 1.56 points. These experiments evaluate how the reranker uses a scalar height input, not a complete GNSS-denied altitude-estimation chain. Every experiment in this paper replays a logged flight, with relative height constructed offline and heading taken from the recorded data; no end-to-end GNSS-denied operation is experimentally demonstrated, and validating the onboard sensing chain remains future work.

The current output is the discrete center of a 317×317 m satellite tile, and the two test sequences primarily assess large-area global candidate selection and short-term trajectory consistency. The pooled metrics are dominated by subset 02 (1071 of 1215 frames), so they should not be read as broad geographic generalization; per-sequence results are reported alongside, and a region-swapped evaluation is reported in Appendix E. These results do not constitute continuous-coordinate estimation or cross-season generalization, nor a complete GNSS-denied navigation system: the paper evaluates a reranking stage whose heading, height, and speed inputs are assumed available from standard onboard sources. In a complete localization pipeline, TRACE-Loc can first establish a stable global candidate, after which local feature matching and geometric estimation can recover a sub-tile position [6–8]. Further evaluation across additional regions, seasons, repeated flights, and real onboard sensors will be needed to characterize this combined pipeline in flight.

The split of the reranking gain between temporal selection and historical projection is itself region-dependent. On the primary test regions, the no-projection recursion barely improves on single-frame retrieval (+0.25 points C@1): subset 02 mixes high-coverage land frames with open-water frames whose top-50 lists usually contain no strong positive, so path competition among retrieval candidates alone adds little, and recovery on difficult frames comes from projected candidates. On the swapped regions (Appendix E), retrieval is uniformly stronger, temporal selection alone contributes +9.12 points, and projection adds a further +2.48 points. This is consistent with the coverage analysis of Section 5.4: projection pays off exactly where the current candidate list intermittently loses the strong positives, so its share of the total gain shrinks as framewise coverage becomes uniformly high. The absolute reranking gain therefore transfers across regions more robustly than any fixed attribution of that gain between temporal selection and projection.

7. Conclusions

This paper presented TRACE-Loc, which decomposes large-area UAV–satellite localization without a known initial position into global candidate generation and causal trajectory selection. The recursion without historical projection is a beam-width-limited Viterbi decode over the retrieval candidates; the contribution of the method is its causal historical candidate projection. The full reranker raises framewise C@1 by 9.66 $\pm$ 2.69 percentage points over single-frame retrieval across three independent training runs (48.67 $\pm$ 0.41% to 58.33 $\pm$ 2.84%), and the projection stage contributes +8.07 $\pm$ 3.75 points of that gain over the identical recursion without projection. On the first training run, whose outputs are used for all detailed comparisons, TRACE-Loc raises C@1 from 49.05% to 61.23%, reduces median error from 150.92 to 93.75 m, and increases D $<$ 300 m from 53.66% to 70.62%, against a tile-quantization error floor of 48.27 m median. The same reranker improves all six retrieval frontends. The per-start cold-start evaluation shows a higher probability of forming a stable position early than single-frame retrieval at every threshold, while also exposing a recovery tail caused by persistent incorrect historical states; against the particle filter, early-fix probabilities are statistically indistinguishable and long-horizon recovery is slower, so the particle filter remains preferable when continuous priors and longer observation windows are available.

Candidate-level temporal reasoning can therefore serve as an online stage between global cross-view retrieval and continuous state estimation. It leaves the retrieval frontend unchanged and does not require an initial position, instead using short-term motion relations over a finite candidate set to improve the consistency of framewise output. The present estimates remain quantized to satellite-tile centers, and the study evaluates a reranking stage rather than a complete GNSS-denied navigation system: its inputs are assumed available from standard onboard sources, and only two geographically held-out sequences are tested. Future work will combine confidence-driven history reset and reacquisition with local geometric refinement and onboard sensing, allowing the initial global fix to converge to a continuous position.

Author Contributions: Conceptualization, H.C. and S.L.; methodology, H.C.; software, H.C.; validation, H.C., Z.L. (Zehao Li) and Z.L. (Zhiguo Liu); formal analysis, H.C.; investigation, H.C.; resources, S.L. and Z.L. (Zhiguo Liu); data curation, H.C. and Z.L. (Zehao Li); writing—original draft preparation, H.C.; writing—review and editing, S.L., Z.L. (Zehao Li) and Z.L. (Zhiguo Liu); visualization, H.C.; supervision, S.L. and Z.L. (Zhiguo Liu); project administration, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The UAV-VisLoc dataset analyzed in this study is publicly available from the source cited in Reference [18]. All derived artifacts used to compute the reported numbers are released as a public Zenodo archive (DOI: 10.5281/zenodo.22900194): the exact split lists for the primary and region-swapped protocols, the VC-LoRA training configuration, the causal visual-speed estimates, the per-start cold-start records for all five methods and thresholds, the per-frame outputs behind every table, and all aggregated statistics with the validation grid. Further requests may be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AHRS	Attitude and Heading Reference System
DEM	Digital Elevation Model
ENU	East–North–Up
FAISS	Facebook AI Similarity Search
GNSS	Global Navigation Satellite System
ORB	Oriented FAST and Rotated BRIEF
RANSAC	Random Sample Consensus
UAV	Unmanned Aerial Vehicle
VO	Visual Odometry
VPR	Visual Place Recognition
VC-LoRA	View-Conditioned Low-Rank Adaptation
TTFF	Time to First Fix

Appendix A. Baseline Configurations, Data Statistics, and Reset Counts

This appendix documents every temporal baseline of Table 1 at the level required for reimplementation. All baselines receive the same per-frame top-50 candidates, timestamps, headings, and visual-speed observations in chronological order. The belief-based baselines use the validation-selected speed-deviation scale ($\sigma_v = 8.0\,\mathrm{m/s}$); the particle filter and forward CV-KF use their own noise scales and take no σ_v. All use the same 20 m/s fallback speed when a causal speed observation is invalid, and the same 10 s rule that clears temporal state across a longer observation gap.

Causal score averaging. The output at frame i is the tile with the highest similarity averaged over a causal window $W = 9$ of rank-aligned candidate lists; no motion information is used.

Belief w/o projection (beam Viterbi). The recursion of Equations (16)–(22) with $K_{\mathrm{nb}} = 0$: node costs are negative cosine similarities, transition costs follow Equations (17)–(19) with $\alpha_h = 3.0$, $\beta_v = 0.2$, $\lambda_T = 0.008$ (validation-selected over $\{0.004, \ldots, 0.05\}$), beam width $B_{\max} = 80$, history $H = 3$, the same frozen configuration as the full method. The forward recursion with per-frame argmin output is identical to an HMM Viterbi decode over the candidate trellis with beam pruning.

Forward constant-velocity Kalman filter. State $[x, y, \dot{x}, \dot{y}]^\top$; constant-velocity motion model with process noise $q = \sigma_a^2$ per axis, $\sigma_a = 4\,\mathrm{m/s^2}$; position measurements $[x, y]$ with $\sigma_m = 250\,\mathrm{m}$; candidate association by appearance plus $0.1\times$ Mahalanobis distance with a χ^2 gate of 25; initialized from the first candidate with heading-aligned velocity at the fallback speed and velocity standard deviation 15 m/s; the predicted velocity is blended with the VO velocity by weight 0.5 before each prediction; parameters validation-selected over a 54-cell grid.

Particle filter. 256 particles in continuous ENU coordinates; at the first frame and after each reset, the particles are drawn from the current top-50 candidate centers by systematic resampling of the appearance weights, with Gaussian jitter of standard deviation 20 m; prediction adds heading-aligned VO displacement plus Gaussian process noise with $\sigma_p = 6\,\mathrm{m/s}$; candidate weights at each frame combine the appearance logits (temperature 0.05) with the mean Gaussian kernel ($\sigma_m = 200\,\mathrm{m}$) of the predicted particle cloud around each candidate; systematic resampling; the output is the top-50 candidate with the highest posterior weight; parameters validation-selected over an 18-cell grid.

Data statistics. The test VO file contains 1213 adjacent frame pairs. Their intervals are 3 s (596 pairs), 5 s (355), 6 s (145), 4 s (89), 2 s (9), and 7 s (3); the remaining pairs span 38–47 s. The 10 s rule triggers 16 state resets in total (9 in subset 02, 7 in subset 10). Sequence durations follow from the same timestamps: subset 02 spans 80.3 min over 1071 frames and subset 10 spans 11.3 min over 144 frames. The strict cold-start evaluation generates

1215 starts per method per threshold. Per-threshold event/censored counts: single frame 592/623, 1178/37, 1188/27; no-projection (unified, $K_{\text{nb}} = 0$, $B_{\max} = 80$) 620/595, 1191/24, 1204/11; CV-KF 566/649, 1181/34, 1184/31; particle filter 777/438, 1190/25, 1199/16; TRACE-Loc 691/524, 1154/61, 1160/55 at 50/100/300 m respectively; the released per-start records contain the full detail.

Table A1. Per-subset cold-start P_{30} (%) by method and distance threshold.

Method	Subset 02 (1071 Starts)			Subset 10 (144 Starts)		
	50 m	**100 m**	**300 m**	**50 m**	**100 m**	**300 m**
Single frame	7.32	51.90	60.12	2.13	18.61	35.17
Belief w/o projection	8.54	56.19	62.96	10.19	52.05	55.41
Forward CV-KF	8.35	53.36	61.54	5.74	36.67	47.40
Particle filter	9.48	57.71	63.83	8.67	51.28	59.83
TRACE-Loc	9.10	58.15	64.27	8.64	51.30	54.67

Appendix B. Symbol Table

Table A2. Symbols used in the main text.

Symbol	Meaning
K_{ret}	retrieval candidates per frame (top-K_{ret})
K	evaluation rank in C@K (metric cutoff)
K_{nb}	historical states projected per retained time step
$B_{\max}$, B_i	beam capacity; number of retained states at frame i
H	number of historical state sets used for projection
λ_T	transition-cost weight
α_h, β_v	heading and speed weights inside E_{trans}
σ_v	speed-deviation scale of E_{vel}
η, τ_{proj}	upper bound and decay of the projected-candidate score
γ	cumulative-cost discount ($\gamma = 1$ in the main configuration)
v_{fb}	fallback speed when VO is invalid (20 m/s)
ε_0	stationary-pair threshold of E_{head} (10^{-6} m)
ε_d	divisor floor of E_{vel} (10^{-3} m)
L, ϵ	consecutive-frame count and threshold of first stable fix
P_{30}, RMST_{600}	KM fix probability by 30 s; restricted mean waiting time to 600 s
R@50	fraction of frames with a strong positive in the top-50 retrieval list

Appendix C. Data Split Membership

Subset-holdout split of the 11 UAV-VisLoc regions: training {01 (800 frames), 03 (768), 04 (738), 09 (766), 11 (590), 08 first 733 frames} = 4395 frames; validation {05 (472), 06 (252), 08 remaining 300} = 1024 frames; test {02 (1071), 10 (144)} = 1215 frames. Training and validation frames of subset 08 share the regional satellite map but do not overlap in frames; neither test region is used for training or for any parameter selection. The region-swapped evaluation of Appendix E exchanges the roles: test $\{05, 06\}$, validation $\{02, 10\}$, training unchanged.

Appendix D. Validation Re-Selection Under the Corrected Transition Cost

Section 3.4 corrects the heading term of E_{trans} for stationary candidate pairs (Equation (17)). To keep every reported number tied to one frozen method definition, all reported experiments retain the configuration that was validation-selected in the original submission ($H = 3$, $\lambda_T = 0.008$, $\eta = 0.4$, $\tau_{\text{proj}} = 0.25$); this combination is itself a member of the 54-cell validation grid. Re-running the full grid under the corrected cost yields a validation argmax of $H = 8$, $\lambda_T = 0.012$, $\eta = 0.4$, $\tau_{\text{proj}} = 0.25$ with vali-

dation C@1 78.32% (the frozen configuration achieves 77.93% under the same corrected code). Evaluated on test, the re-selected variant reaches 60.82% C@1 (median 92.35 m, D $<$ 300 m 70.78%): its validation advantage over the frozen configuration is $+0.39$ points (95% CI $[-0.39, +1.27]$, paired moving-block bootstrap over the 1024 validation frames), i.e., statistically indistinguishable, so it confers no test advantage (60.82% vs. 61.23%). Both parameterizations are reported; the frozen one is used throughout.

Appendix E. Region-Swapped Evaluation

To probe role-swap consistency without retraining, the split roles of the two test regions and the two previously unused regions are exchanged: test $\{05, 06\}$ (724 frames), validation $\{02, 10\}$ (1215 frames), training unchanged. The canonical checkpoint is retained: it was saved at a fixed epoch (60) by the checkpoint schedule, and the per-epoch validation loss was logged but never used for model selection, so no exchanged region influenced training or checkpoint choice. Regions 05 and 06 did, however, serve as the development validation split for the belief hyperparameters of the primary evaluation, so this experiment probes split robustness rather than fully novel terrain. Belief hyperparameters and σ_v are re-selected on the swapped validation (54-cell grid $\times\sigma_v \in \{4, 6, 8\}$; selected $\sigma_v = 6.0$, $H = 8$, $\lambda_T = 0.008$, $\eta = 0.2$, $\tau_{\text{proj}} = 0.35$) before the swapped test is touched. On the swapped test regions, single-frame retrieval achieves 67.68% C@1 (median 77.04 m), and TRACE-Loc reaches 79.28% C@1, 65.26 m median, and 91.30% D $<$ 300 m (+11.60 points C@1); per-region, subset 05 reaches 76.69% C@1 (median 71.27 m) and subset 06 reaches 84.13% (median 56.06 m). The mechanism split, however, reverses relative to the primary test set: the no-projection control already reaches 76.80% C@1 (median 67.82 m), so on these regions pure temporal selection contributes +9.12 points and projection adds a further +2.48 points, whereas on the primary test set temporal selection contributes +0.25 points and projection +11.93 points (Section 5.4). The overall reranking gain transfers at comparable strength on both splits (+11.60 and +12.18 points over single-frame retrieval), but the split of that gain between temporal selection and projection is region-dependent; Section 6.3 discusses why.

References

1. Arandjelović, R.; Gronat, P.; Torii, A.; Pajdla, T.; Sivic, J. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2016; pp. 5297–5307. [CrossRef]
2. Ali-bey, A.; Chaib-draa, B.; Giguère, P. MixVPR: Feature mixing for visual place recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*; IEEE: New York, NY, USA, 2023; pp. 2997–3006. [CrossRef]
3. Izquierdo, S.; Civera, J. Optimal transport aggregation for visual place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2024; pp. 17658–17668. [CrossRef]
4. Deuser, F.; Habel, K.; Oswald, N. Sample4Geo: Hard negative sampling for cross-view geo-localisation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: New York, NY, USA, 2023; pp. 16801–16810. [CrossRef]
5. Siméoni, O.; Vo, H.V.; Seitzer, M.; Baldassarre, F.; Oquab, M.; Jose, C.; Khalidov, V.; Szafraniec, M.; Yi, S.; Ramamonjisoa, M.; et al. DINOv3. *arXiv* **2025**, arXiv:2508.10104. [CrossRef]
6. Sarlin, P.-E.; DeTone, D.; Malisiewicz, T.; Rabinovich, A. SuperGlue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2020; pp. 4937–4946. [CrossRef]
7. Lindenberger, P.; Sarlin, P.-E.; Pollefeys, M. LightGlue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: New York, NY, USA, 2023; pp. 17581–17592. [CrossRef]
8. Zhou, X.; Zhang, X.; Yang, X.; Zhao, J.; Liu, Z.; Shuang, F. Towards UAV localization in GNSS-denied environments: The SatLoc dataset and a hierarchical adaptive fusion framework. *Remote Sens.* **2025**, *17*, 3048. [CrossRef]
9. Milford, M.J.; Wyeth, G.F. SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*; IEEE: New York, NY, USA, 2012; pp. 1643–1649. [CrossRef]

10. Wang, Z.; Shi, D.; Qiu, C.; Jin, S.; Li, T.; Shi, Y.; Liu, Z.; Qiao, Z. Sequence matching for image-based UAV-to-satellite geolocalization. *IEEE Trans. Geosci. Remote Sens.* **2024**, *62*, 1–15. . [CrossRef]
11. Jurevičius, R.; Marcinkevičius, V.; Šeibokas, J. Robust GNSS-denied localization for UAV using particle filter and visual odometry. *Mach. Vis. Appl.* **2019**, *30*, 1181–1190. [CrossRef]
12. Xia, Z.; Booij, O.; Manfredi, M.; Kooij, J.F.P. Cross-view matching for vehicle localization by learning geographically local representations. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5921–5928. [CrossRef]
13. Kinnari, J.; Verdoja, F.; Kyrki, V. GNSS-denied geolocalization of UAVs by visual matching of onboard camera images with orthophotos. In *Proceedings of the 20th International Conference on Advanced Robotics (ICAR)*; IEEE: New York, NY, USA, 2021; pp. 555–562. [CrossRef]
14. Bianchi, M.; Barfoot, T.D. UAV localization using autoencoded satellite images. *IEEE Robot. Autom. Lett.* **2021**, *6*, 1761–1768. [CrossRef]
15. Zheng, Z.; Wei, Y.; Yang, Y. University-1652: A multi-view multi-source benchmark for drone-based geo-localization. In *Proceedings of the 28th ACM International Conference on Multimedia*; ACM: New York, NY, USA, 2020; pp. 1395–1403. [CrossRef]
16. Dai, M.; Zheng, E.; Feng, Z.; Qi, L.; Zhuang, J.; Yang, W. Vision-based UAV self-positioning in low-altitude urban environments. *IEEE Trans. Image Process.* **2024**, *33*, 493–508. [CrossRef] [PubMed]
17. Zhu, S.; Shah, M.; Chen, C. TransGeo: Transformer is all you need for cross-view image geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2022; pp. 1162–1171. [CrossRef]
18. Xu, W.; Yao, Y.; Cao, J.; Wei, Z.; Liu, C.; Wang, J.; Peng, M. UAV-VisLoc: A large-scale dataset for UAV visual localization. *arXiv* **2024**, arXiv:2405.11936. [CrossRef]
19. Li, J.; Sun, Y.; Xiang, Y.; Lei, L. One-to-many retrieval between UAV images and satellite images for UAV self-localization in real-world scenarios. *Remote Sens.* **2025**, *17*, 3045. [CrossRef]
20. Ji, Y.; He, B.; Tan, Z.; Wu, L. MMGeo: Multimodal compositional geo-localization for UAVs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: New York, NY, USA, 2025; pp. 25165–25175. [CrossRef]
21. Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.V.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; et al. DINOv2: Learning robust visual features without supervision. *arXiv* **2024**. [CrossRef]
22. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. LoRA: Low-rank adaptation of large language models. In Proceedings of the International Conference on Learning Representations (ICLR), Virtual, 25–29 April 2022. Available online: https://openreview.net/forum?id=nZeVKeeFYf9 (accessed on 11 August 2026).
23. Zhong, Z.; Tang, Z.; He, T.; Fang, H.; Yuan, C. Convolution meets LoRA: Parameter efficient finetuning for segment anything model. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024. Available online: https://openreview.net/forum?id=ezscMer8L0 (accessed on 11 August 2026).
24. Wang, R.; Yuan, W.; Yuan, W.; Liu, T.; Xi, X.; Zhu, Y. GenGeo: Robust cross-view geo-localization via foundation model and dynamic feature aggregation. *Remote Sens.* **2026**, *18*, 1116. [CrossRef]
25. Sun, J.; Shen, Z.; Wang, Y.; Bao, H.; Zhou, X. LoFTR: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: New York, NY, USA, 2021; pp. 8918–8927. [CrossRef]
26. Zhou, Q.; Tang, H.; Zhang, Z.; Xu, Y.; Xiao, F.; Jia, Y. A hierarchical absolute visual localization system for low-altitude drones in GNSS-denied environments. *Remote Sens.* **2025**, *17*, 3470. [CrossRef]
27. Deng, Z.; Case, A.; Sentis, L. AirLock+: Scaling UAV-to-satellite image registration for target geolocalization and geospatial augmented reality. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*; IEEE: New York, NY, USA, 2026; pp. 3340–3349. [CrossRef]
28. Zhang, X.; Sultani, W.; Wshah, S. Cross-view image sequence geo-localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*; IEEE: New York, NY, USA, 2023; pp. 2913–2922. [CrossRef]
29. Shpagin, P.; Panchenko, T. NaviLoc: Trajectory-level visual localization for GNSS-denied UAV navigation. *Drones* **2026**, *10*, 97. [CrossRef]
30. Radenović, F.; Tolias, G.; Chum, O. Fine-tuning CNN image retrieval with no human annotation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 1655–1668. [CrossRef] [PubMed]
31. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*; IEEE: New York, NY, USA, 2011; pp. 2564–2571. [CrossRef]
32. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395. [CrossRef]
33. Kaplan, E.L.; Meier, P. Nonparametric estimation from incomplete observations. *J. Am. Stat. Assoc.* **1958**, *53*, 457–481. [CrossRef]
34. Royston, P.; Parmar, M.K.B. Restricted mean survival time: An alternative to the hazard ratio for the design and analysis of randomized trials with a time-to-event outcome. *BMC Med. Res. Methodol.* **2013**, *13*, 152. [CrossRef] [PubMed]

35. Künsch, H.R. The jackknife and the bootstrap for general stationary observations. *Ann. Stat.* **1989**, *17*, 1217–1241. [CrossRef]
36. Politis, D.N.; Romano, J.P. A circular block bootstrap procedure for stationary data. In *Proceedings of the IEEE International Conference on Computing and Information (ICCI)*; IEEE: New York, NY, USA, 1992; pp. 185–188.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.