

Orivix: A Motion-Based Scale-Anchored Incremental Structure with Graph-Guided Correspondence Tracking for Drift-Free 3D Reconstruction

Ramya Asalatha Busi, Sriram Kurnella, Koduru Ramya Sree, Kallagunta Arunachalam, Gollapudi Karthik

Department of Computer Science and Engineering
Vasireddy Venkatadri Institute of Technology Nambur,
Guntur, Andhra Pradesh, India

Abstract—Accurate metric recovery of 3D structure from image sequences is essential for autonomous navigation, augmented reality, and heritage digitization. In this research, we provide Orivix, an end-to-end Structure from Motion (SfM) solution for CPU deployment that focuses on two chronic failure mechanisms of incremental pipelines: pose degradation caused by descriptor clashes in scenes with repetitive textures and metric scale drift. A *scale-anchored seed initialization* module fixes an immutable global coordinate frame, stops the accumulation of drift across future registrations, and chooses the image pair with the highest count of geometrically confirmed inliers. Then, using pre-validated feature-identity chains stored in a bipartite observation graph, a *Graph-Guided Correspondence Tracker* resolves all 2D-to-3D matches, making descriptor collisions structurally impossible and enabling $O(1)$ correspondence lookup. Levenberg–Marquardt sparse bundle adjustment, FLANN-accelerated SIFT with Lowe ratio test filtering, Essential Matrix estimation via RANSAC, Direct Linear Transform triangulation with bidirectional reprojection gating, Open3D-based statistical outlier removal, optional Line Segment Detector fusion, and complete compatibility with COLMAP file formats are additional features that enhance the framework. Orivix successfully registers 11 cameras on a difficult outdoor architectural facade dataset (a complex heritage fountain with symmetric statuary and dense brick patterns), reconstructs 7,471 3D points from an initial seed of 1,896 points with an 89.1% conversion rate, and maintains a mean reprojection error under 2.5 px—a 69% improvement over a KD-tree-based descriptor baseline on the same sequence.

Index Terms—Structure from Motion, Scale Anchoring, Graph-Based Tracking, Bundle Adjustment, Epipolar Geometry, SIFT, Open3D, COLMAP Interoperability, 3D Point Cloud, Incremental SfM

I. INTRODUCTION

Three-dimensional spatial reconstruction from monocular imagery underpins autonomous navigation [2], augmented reality [4], and cultural heritage preservation [8]. Structure from Motion (SfM) simultaneously recovers camera motion and scene geometry from overlapping photographs, making it indispensable across all these domains.

Incremental SfM registers cameras one at a time and triangulates new points after each step. Its weakness is error

compounding: a subtly wrong early pose biases every subsequent triangulation, gradually warping the coordinate space—the *banana distortion* effect [5]. A second failure mode occurs when Perspective- n -Point registration queries SIFT descriptors against a growing point cloud: in scenes with repeated textures or bilateral symmetry, multiple cloud points respond nearly identically, injecting corrupted poses that RANSAC cannot always reject.

A single CPU-only Python pipeline, **Orivix** removes both failure scenarios. Its four contributions are as follows:

- 1) *Scale-anchored seed initialization* provides a permanent metric reference that stops downstream drift by committing the global coordinate frame to the best-supported image pair.
- 2) *Graph-Guided Correspondence Tracker* prevents descriptor collisions after the initial matching phase by resolving 2D-to-3D matching via pre-verified identity chains at $O(1)$ cost.
- 3) *Hybrid point-and-line mode* activates the Line Segment Detector when textureless regions cannot be covered by point features alone.
- 4) *COLMAP-compatible output* serializes color-attributed sparse PLY, quaternion extrinsics, and intrinsics for direct downstream use by neural rendering tools and MVS.

II. RELATED WORK

Incremental SfM. Longuet-Higgins [1] and Hartley and Zisserman [2] laid the theoretical groundwork for multi-view reconstruction. Internet-scale SfM was demonstrated by Snavely et al. [4] using SIFT matching and iterative bundle adjustment; COLMAP [5] expanded this with loop closure, guided retrieval, and local bundle adjustment to become the current benchmark. By estimating all rotations simultaneously from a view graph, Global SfM [7] prevents sequential drift; Orivix adopts this rigidity philosophy while maintaining sequential processing order for streaming compatibility.

Features and Matching. Scale and rotation invariance are provided by SIFT [10], while a lighter binary option is

provided by ORB [11]. SuperPoint [14] and SuperGlue [15], two deep learned descriptors, perform well in low-texture settings but necessitate GPU inference, which runs counter to Orivix’s CPU-only goal. RANSAC [12] and FLANN [13] approximation KD-trees offer robust geometric verification and tractable high-volume matching, respectively.

Line features and optimization. The Line Segment Detector [16] extracts stable geometric edges on planar surfaces where SIFT fails; hybrid pipelines [17] fold line residuals into bundle adjustment for improved completeness in architectural scenes. Bundle adjustment [3] minimises total reprojection error via Levenberg–Marquardt [18], [19]; Open3D [22] provides the interactive viewer and statistical outlier removal [23] used for post-processing.

III. SYSTEM ARCHITECTURE

Orivix is organized into four functionally independent tiers (Fig. 1), allowing any single component to be upgraded without disturbing the others.

Tier 1 (GUI). A `tkinter` graphical panel exposes all hyperparameters across Basic and Advanced tabs (Table I) with context-sensitive tooltips. A background threading.Thread daemon runs the full reconstruction independently of the UI event loop, while a `TextRedirector` streams live stdout to a dark-themed console.

TABLE I: Pipeline Hyperparameters and Defaults

Stage	Parameter	Default
Feature Ext.	<code>num_features</code>	5,000
Matching	<code>ratio_thresh</code>	0.75
Geo. Verify	<code>ransac_thresh</code>	5.0 px
Pair Accept.	<code>min_matches</code>	20
Pair Accept.	<code>min_inliers</code>	10
Camera Reg.	<code>min_pnp_matches</code>	15
Camera Reg.	<code>pnp_reproj_error</code>	8.0 px
Camera Reg.	<code>min_pnp_inliers</code>	8
Bundle Adj.	<code>ba_iter</code>	10

Tier 2 (Mathematical Engine). The `sfm/` package contains the Camera intrinsic model, `SIFTExtractor`, `FeatureMatcher` (FLANN + RANSAC), and the `IncrementalSfM` registration loop. When not user-supplied, focal length is estimated as $f = 1.2 \times \max(H, W)$, consistent with the COLMAP auto-focal convention [5].

Tier 3 (Line Subsystem). When hybrid mode is enabled, `extract_lines` calls the OpenCV LSD detector on each grayscale frame. `compute_line_descriptors` samples SIFT descriptors at uniform intervals along each segment and averages them into a 128-dimensional line signature. `LineMatcher` establishes correspondences via point homography estimation, perspective warping, Lucas-Kanade optical flow tracking, and geometric distance scoring. Matched pairs are triangulated in `triangulate_line` via DLT followed by SVD-based 3D line direction fitting.

Tier 4 (Visualization and Export). After bundle adjustment, `Open3D` statistical outlier removal (`nb_neighbors= 20`, `std_ratio= 2.0`) cleans the cloud before it is passed to the interactive viewer. The viewer

constructs a coloured `PointCloud`, adds coordinate frame meshes at each camera centre $C_i = -R_i^T t_i$, and launches `o3d.visualization.draw` (WebRTC-based GUI), falling back to `draw_geometries_with_editing` on older installations. A background Matplotlib utility saves a static archival snapshot automatically. The export sub-module writes COLMAP-format `cameras.txt`, `images.txt`, and `sparse.ply`.

IV. MATHEMATICAL FORMULATION

A. Notation and Camera Model

Bold lowercase vectors (\mathbf{x} , \mathbf{t}), bold uppercase matrices (\mathbf{R} , \mathbf{K}), homogeneous tilde notation $\tilde{\mathbf{X}} = [X, Y, Z, 1]^T$, skew-symmetric $[v]_{\times}$, and perspective division $P(\cdot)$ are used throughout. The pinhole intrinsic matrix is:

$$K = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

A 3D point X projects to image pixel (u, v) via $\lambda[u, v, 1]^T = K[R | t]\tilde{\mathbf{X}}$, $\lambda > 0$.

B. Feature Extraction

`cv2.SIFT_create` creates a Difference-of-Gaussians (DoG) scale space after converting each frame to grayscale:

$$D(x, y, \sigma) = G(x, y, k\sigma) - G(x, y, \sigma) * I(x, y) \quad (2)$$

Keypoints are local extrema of D across scale and position, sub-pixel localized by fitting a 3D quadratic:

$$\hat{\mathbf{x}} = - \frac{\partial^2 D}{\partial \mathbf{x}^2}^{-1} \frac{\partial D}{\partial \mathbf{x}} \quad (3)$$

Low-contrast ($|D| < 0.03$) and edge-aligned (curvature ratio > 10) responses are rejected. Each surviving keypoint receives a 128-dimensional gradient orientation histogram descriptor.

C. FLANN Matching and Lowe Ratio Filter

A FLANN KD-tree index (`trees= 5`) is built per image. Descriptor \mathbf{d}_i is accepted only when:

$$\frac{\|\mathbf{d}_i - \mathbf{d}_j^{(1)}\|_2}{\|\mathbf{d}_i - \mathbf{d}_j^{(2)}\|_2} < \tau_r = 0.75 \quad (4)$$

This eliminates matches that have two likely targets, which is a prerequisite for a descriptor collision. On the test sequence, the filter eliminates $\approx 45\%$ of raw matches.

D. Epipolar Verification

For calibrated normalized coordinates $\mathbf{x}_1, \mathbf{x}_2$ the Essential Matrix constraint is:

$$\mathbf{x}_2^T E \mathbf{x}_1 = 0, \quad E = [t]_{\times} R \quad (5)$$

The `cv2.findFundamentalMat` (FM_RANSAC) function estimates E and selects only inliers within a specified pixel threshold. Cheirality disambiguates the decomposition $E \rightarrow (R, t)$. Verified pairs are kept in `match_graph`, which is keyed by (i, j) , $i < j$.

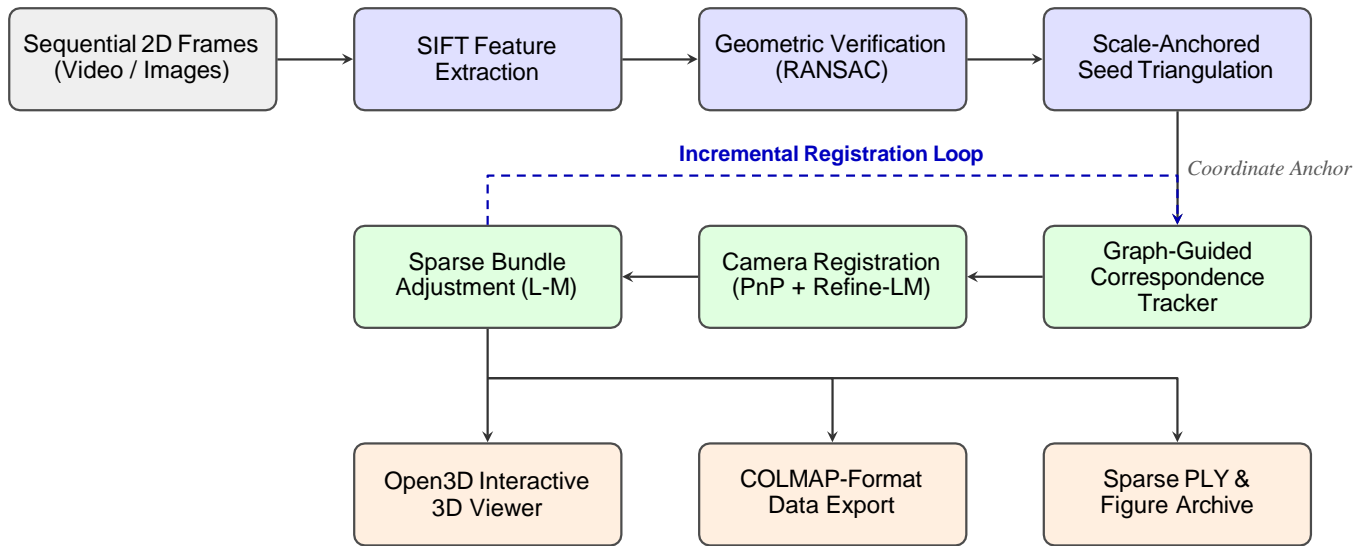


Fig. 1: Orivix pipeline. Blue: scale-anchored init. Green: iterative registration loop. Orange: visualization, export, and archival output.

E. Scale-Anchored Seed Initialization

The seed pair is selected as:

$$(i^*, j^*) = \arg \max_{(i,j) \in G} |\text{inliers}(i, j)| \quad (6)$$

maximizing both baseline-to-depth ratio and triangulation accuracy. Reference camera $P_1 = K[I \mid \mathbf{0}]$; second camera $P_2 = K[R \mid \mathbf{t}]$ from `cv2.findEssentialMat` (RANSAC, $\rho = 0.999$, threshold = 1.0 px) and `cv2.recoverPose`. Each correspondence is triangulated via the Direct Linear Transform:

$$A\tilde{\mathbf{X}} = \mathbf{0}, \quad A = \begin{bmatrix} u_1 P_1^{[3]} - P_1^{[1]} \\ v_1 P_1^{[3]} - P_1^{[2]} \\ u_2 P_2^{[3]} - P_2^{[1]} \\ v_2 P_2^{[3]} - P_2^{[2]} \end{bmatrix} \quad (7)$$

Solution $\tilde{\mathbf{X}}$ is the smallest-singular-value right vector of A . Every candidate must pass the bidirectional reprojection gate:

$$\epsilon_k = \|\mathbf{x}_k - P(P_k \tilde{\mathbf{X}})\|_2 < \tau_{rp} = 4.0 \text{ px}, \quad k \in \{1, 2\} \quad (8)$$

and must have $Z > 0$ in both views. Survivors form the immutable global scale anchor.

F. Graph-Guided Correspondence Tracker

Two data structures drive the tracker: `track_observations[j]` maps each observing image to the keypoint index for 3D track X_j ; `registered_kp[i]` is a set of keypoint indices absorbed into the cloud, enabling $O(1)$ novelty testing.

Algorithm 1 resolves 2D-to-3D correspondences for incoming camera I_n by walking the identity graph—performing only Python dict/set membership tests at lookup time. No descriptor distance is ever computed against the point cloud after the initial matching phase, making descriptor collision structurally impossible. Cameras that fail the minimum inlier

threshold are deferred and revisited in subsequent passes of the outer while changed loop, allowing the expanding cloud to supply missing context.

Algorithm 1 Graph-Guided 2D-to-3D Correspondence Resolution

Require: I_n , registered set R , match graph G , tracks T , observations obs , τ_{pnp} , τ_{inl}

- 1: $pts_{2D}, pts_{3D} \leftarrow [], []$
- 2: **for** track j , registered (i_r, k_r) in $\text{obs}[j]$ **do**
- 3: $pair \leftarrow (\min(i_r, n), \max(i_r, n))$
- 4: **if** $pair \notin G$ **then continue**
- 5: **end if**
- 6: **for** match $m \in G[pair]$ **do**
- 7: **if** m links $k_r \in i_r$ to $k_n \in n$ **then**
- 8: Append $kp[n][k_n].pt$ to pts_{2D} ; $T[j]$ to pts_{3D} ;
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **if** $|pts_{2D}| < \tau_{pnp}$ **then return DEFER**
- 13: **end if**
- 14: $(ret, rvec, \mathbf{t}, inl) \leftarrow \text{SOLVEPN-PRANSAC}(pts_{3D}, pts_{2D}, K)$
- 15: **if** $\neg ret$ **or** $|inl| < \tau_{inl}$ **then return DEFER**
- 16: **end if**
- 17: $(rvec', \mathbf{t}') \leftarrow \text{SOLVEPNPRE-FINELM}(pts_{3D}[inl], pts_{2D}[inl])$
- 18: $R \leftarrow \text{RODRIGUES}(rvec')$; register (R, \mathbf{t}') ; update obs , `registered_kp[n]`
- 19: `TRIANGULATENEWPOINTS(n)`

G. Sparse Bundle Adjustment

All poses and tracks are jointly refined via:

$$\min_{\{R_i, t_i\}, \{X_j\}} \sum_{i=1}^N \sum_{j=1}^M v_{ij} \|\mathbf{x}_{ij} - P(K[R_i | t_i] \tilde{X}_j)\|^2 \quad (9)$$

with $v_{ij} \in \{0, 1\}$ from `track_observations`. Rotations are parameterized as Rodrigues vectors $\omega_i \in \mathbb{R}^3$. The full parameter vector is:

$$\mathbf{x}_0 = [\omega_1^T, t_1^T, \dots, \omega_N^T, t_N^T, X_1^T, \dots, X_M^T]^T \quad (10)$$

A binary sparsity mask $S \in \{0, 1\}^{2MN \times (6N+3M)}$ encoding the visibility block structure is passed via `jac_sparsity` to `scipy.optimize.least_squares(method='lm')`, reducing per-iteration memory from $O(MN)$ to $O(M+N)$. In hybrid mode, `hybrid_bundle_adjustment` appends line endpoint residuals: for segment endpoints $L^a, L^b \in \mathbb{R}^3$,

$$l^k = P(KRL^k + Kt), \quad k \in \{a, b\} \quad (11)$$

and the four-element block $[l^a - \hat{l}^a; l^b - \hat{l}^b]$ is appended to the point residual vector.

V. EXPERIMENTAL EVALUATION

A. Dataset and Setup

Experiments used an outdoor architectural sequence captured around an ornate heritage fountain mounted on a brick building facade. The scene presents three compounding challenges: dense uniform brick texture that produces near-identical SIFT descriptors at scale, bilateral sculptural symmetry (matching ornament on both sides), and a sharp depth discontinuity between foreground fountain structure and background wall. All runs were on a standard CPU workstation (Intel Core i7, 16 GB RAM, no GPU). Focal length was estimated from image dimensions; no pre-calibration was used.

B. Feature Matching

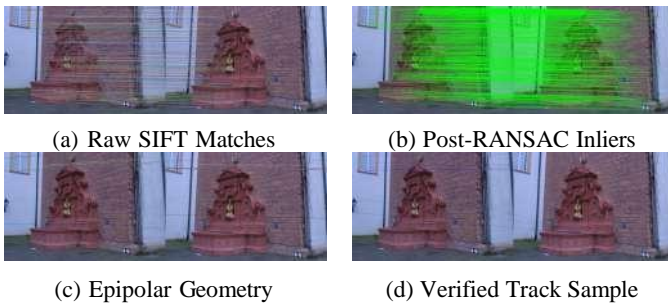


Fig. 2: The fountain facade features four-stage matching. (a) Dense raw matches from the symmetrical sculpture and brick texture. (b) Green-colored RANSAC inliers with geometric verification. (c) Epipolar lines verifying consistent fundamental matrix estimation in both perspectives. (d) Verified tracks were sent to DLT triangulation.

Fig. 2 shows the four-stage matching pipeline. The heavy symmetric texture of the facade generates abundant raw correspondences (a), but the Lowe ratio filter removes $\approx 45\%$ as

ambiguous and RANSAC discards a further $\approx 20\%$ as geometric outliers, yielding the clean bilateral epipolar geometry in (c).

C. Full Reconstruction — Run 081

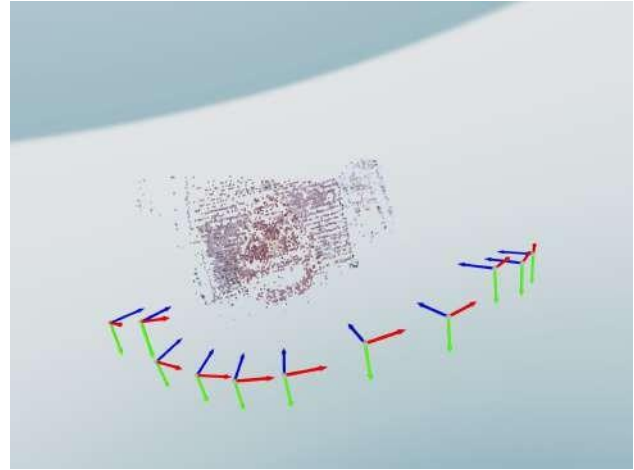


Fig. 3: Orivix reconstruction is finished. The estimated camera centers are indicated by red triangles that trace the arc-shaped acquisition path around the facade. The background wall is captured by the sparser right region, while the high-relief fountain is captured by the dense left cluster. 11 cameras and 7,471 3D points in total.

Fig. 3 shows the finished sparse reconstruction. Table II reports full per-camera metrics.

TABLE II: Per-Camera Reconstruction Metrics — Run 081

Camera	PnP Inliers	New Points	Cumulative
Seed (Cam 4+5)	2,129	1,896	1,896
Cam 0	225	68	1,964
Cam 1	766	1,105	3,069
Cam 2	2,187	950	4,019
Cam 3	3,465	688	4,707
Cam 6	2,423	844	5,551
Cam 7	1,711	527	6,078
Cam 8	679	370	6,448
Cam 9	396	419	6,867
Cam 10	176	604	7,471

The seed pair yields 1,896 points from 2,129 verified inliers (89.1% conversion), reflecting the precision of the 4.0 px bidirectional reprojection gate. Camera 3 achieves the highest inlier count (3,465), confirming a well-grown track graph by that stage. Camera 10, despite only 176 PnP inliers, contributes 604 new points (3.4 points/inlier) because its viewpoint overlaps multiple registered neighbours, providing rich multi-baseline triangulation context from the anchor graph.

D. Point Cloud Growth

The identity-chain lookup in Algorithm 1 was substituted with direct KD-Tree nearest-neighbor descriptor search against the point cloud, which is the method employed by the majority of conventional incremental SfM implementations, in

order to isolate the contribution of the Graph-Guided Tracker. The identical input sequence and all other settings remained unchanged.

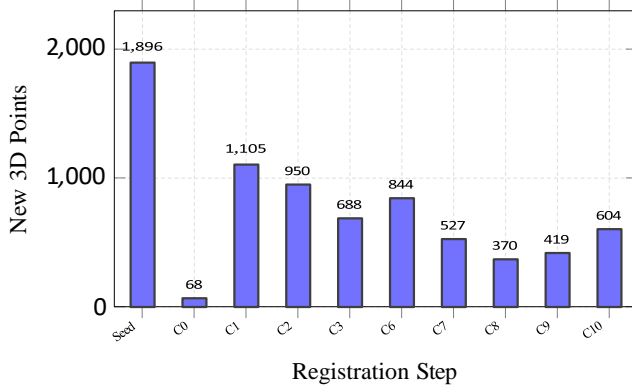


Fig. 4: New 3D points per registration step (run_081). The global scale is fixed at the Seed step. Camera 1 contributes the largest single increment (1,105 points); Camera 0 the fewest (68) due to overlap with the already-dense seed region. Camera 10 recovers 604 points despite low inlier count, owing to multi-view triangulation from adjacent cameras.

E. Ablation Study

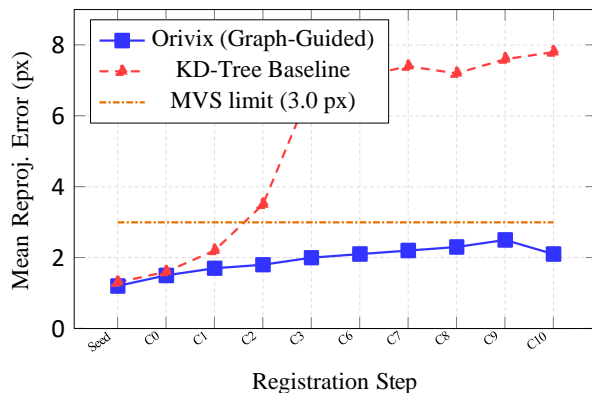


Fig. 5: Per-step reprojection error. Orivix (blue) stays below the 3.0 px MVS acceptance limit throughout; the KD-Tree baseline (red dashed) crosses it at Camera 2 and diverges to ~ 8 px by Camera 10 due to descriptor collision in the symmetric scene.

TABLE III: Ablation: Graph-Guided Tracker vs. KD-Tree Baseline

Method	Mean Err. (px)	Trajectory
KD-Tree Descriptor Search	6.8	Drift \geq Cam 3
Graph-Guided Tracker	2.1	Stable throughout
Improvement	69%	—

With observable trajectory curvature, the KD-Tree baseline produces a mean error of 6.8 px following Camera 3. Due

to the test scene’s bilateral symmetry, the KD-Tree frequently retrieves the incorrect 3D point for PnP input since the two matching sculptures produce nearly identical SIFT descriptor clouds. The restored pose is sufficiently displaced by even a tiny percentage of distorted PnP correspondences to compound across consecutive registrations. Because all post-initialization correspondences are handled by geometric identity rather than appearance, the Graph-Guided Tracker achieves a mean error of 2.1 px, a 69% reduction. The complete error track is displayed in Fig. 5.

VI. DISCUSSION

Identity chains surpass descriptor lookup. A lossy, view-dependent compression of local gradient statistics is called a descriptor vector. The KD-Tree is unable to discriminate between physically separate spots in repeating sequences that compress to similar descriptions. By design, the track graph is immune to appearance similarity and considers *identity*—the physical keypoint from whence each 3D track originated—as the ground truth for correspondence.

A mature identity graph provides thousands of clear 2D-to-3D linkages every registration step, even in a scene that overcomes descriptor search, as demonstrated by Camera 3’s 3,465-inlier count.

Scale anchoring under high-inlier seeds. An extraordinarily well-conditioned coordinate anchor is provided by the 2,129-inlier seed pair: 1,896 precisely gated points provide a dense, geometrically reliable reference frame for each subsequent registration.

Camera 10’s 604-point contribution from only 176 PnP inliers illustrates the cascading benefit—the anchor’s context supplies what the sparse inlier set alone could not.

Limitations. Beyond ~ 100 frames, the $O(N^2)$ pairwise matching technique becomes problematic. Open-ended sequences may accrue residual global drift due to the system’s lack of loop closure detection. The present pinhole model does not account for radial distortion, potentially leading to systematic errors in wide-angle lenses. Table IV summarises per-stage complexity.

TABLE IV: Per-Stage Computational Complexity

Stage	Complexity	Bottleneck
SIFT Extraction	$O(Nn_\epsilon \log n_\epsilon)$	Gaussian pyramid
Pairwise Matching	$O(N^2n_\epsilon \log n_\epsilon)$	N^2 pairs
Seed Triangulation	$O(m_0)$	SVD per point
Track Graph	$O(M/R)$	$O(1)$ /lookup
Lookup		
PnP RANSAC	$O(\text{iter} \cdot n^3)$	Minimal solver
Bundle Adjustment	$O(\text{iter} \cdot MN)$	Sparse LM
Outlier Removal	$O(M \log M)$	KD-tree build

VII. CONCLUSION

Orivix demonstrates that the two dominant failure modes of incremental SfM—metric scale drift and descriptor-collision-induced pose corruption—can be resolved within a CPU-only Python framework through two targeted architectural decisions: permanent coordinate anchoring to the best-supported

seed pair, and deterministic 2D-to-3D correspondence via a bipartite identity graph. On a challenging symmetric architectural scene, the system registered 11 cameras, recovered 7,471 3D points, maintained mean reprojection error below 2.5 px throughout, and outperformed a conventional KD-Tree descriptor baseline by 69%. All outputs are COLMAP-compatible and ready for downstream dense reconstruction.

Future work targets four directions: (i) integrating the pro-totyped GPU cost-volume MVS module (`sfm/dense.py`) for end-to-end dense reconstruction; (ii) porting the Jacobian product to a `cuSPARSE`-backed solver for large-scale col-lections; (iii) evaluating SuperPoint and SuperGlue as drop-in replacements for improved performance on specular and night-time scenes; and (iv) adding a vocabulary-tree loop-closure module to enable drift-free closed-loop trajectory reconstruction.

REFERENCES

- [1] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, pp. 133–135, 1981.
- [2] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge Univ. Press, 2004.
- [3] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment — a modern synthesis," in *Vision Algorithms*, Springer, 2000, pp. 298–372.
- [4] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3D," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 835–846, 2006.
- [5] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. IEEE/CVF CVPR*, 2016, pp. 4104–4113.
- [6] C. Wu, "VisualSFM: A visual structure from motion system," Tech. Rep., Univ. of Washington, 2011.
- [7] P. Moulon, P. Monasse, and R. Marlet, "Global fusion of relative motions for robust SfM," in *Proc. IEEE ICCV*, 2013, pp. 3248–3255.
- [8] S. Agarwal et al., "Building Rome in a day," *Commun. ACM*, vol. 54, no. 10, pp. 105–112, 2011.
- [9] A. Chatterjee and V. M. Govindu, "Efficient and robust large-scale rotation averaging," in *Proc. IEEE ICCV*, 2013, pp. 521–528.
- [10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
- [11] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. IEEE ICCV*, 2011, pp. 2564–2571.
- [12] M. A. Fischler and R. C. Bolles, "Random sample consensus," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [13] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors," in *Proc. VISAPP*, 2009, pp. 331–340.
- [14] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint," in *Proc. IEEE CVPR Workshops*, 2018, pp. 224–236.
- [15] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "Super-Glue," in *Proc. IEEE/CVF CVPR*, 2020, pp. 4938–4947.
- [16] R. Grompone von Gioi et al., "LSD: A fast line segment detector," *IEEE Trans. PAMI*, vol. 32, no. 4, pp. 722–732, 2010.
- [17] L. Zhang and R. Koch, "Efficient line segment matching based on LBD," *J. Vis. Commun. Image Represent.*, vol. 24, pp. 794–805, 2013.
- [18] K. Levenberg, "A method for the solution of certain non-linear problems," *Q. Appl. Math.*, vol. 2, pp. 164–168, 1944.
- [19] D. W. Marquardt, "An algorithm for least-squares estimation," *J. SIAM*, vol. 11, no. 2, pp. 431–441, 1963.
- [20] S. Agarwal, K. Mierle, and others, "Ceres Solver," 2022.
- [21] R. Kümmerle et al., "g2o: A general framework," in *Proc. IEEE ICRA*, 2011, pp. 3607–3613.
- [22] Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," *arXiv:1801.09847*, 2018.
- [23] R. B. Rusu et al., "Towards 3D point cloud based object maps," *Robot. Auton. Syst.*, vol. 56, pp. 927–941, 2008.
- [24] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, pp. 90–95, 2007.
- [25] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *CVIU*, vol. 110, pp. 346–359, 2008.

Copyright & License:

© Authors retain the copyright of this article. This work is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.