COMP0249 CW2 Part 2: Running COLMAP and ORB-SLAM2 on Custom Data Sequences

Amlan Sahoo
MSc Robotics and AI
Department of Computer Science
University College London
amlan.sahoo.24@ucl.ac.uk

Lorenzo Uttini
MSc Robotics and AI
Department of Computer Science
University College London
lorenzo.uttini.24@ucl.ac.uk

Ziya Ruso
MSc Robotics and AI
Department of Computer Science
University College London
ziya.ruso.24@ucl.ac.uk

I. Introduction

In Part II of this coursework, we collected custom data sequences in both indoor and outdoor environments to evaluate the performance of the COLMAP and ORB-SLAM2 systems. These datasets were processed to perform 3D reconstruction using COLMAP, and tracking and mapping using ORB-SLAM2. The resulting camera trajectories were then converted into the TUM format for evaluation using the *evo* tools. We experimented with various shots under different conditions. In this report, we present the best-performing indoor and outdoor sequences and provide a performance comparison between the results of COLMAP and ORB-SLAM2.

II. Data Acquisition

A. Outdoor Dataset

The outdoor dataset was recorded around an abandoned, dilapidated house structure with minimal dynamic elements. The video was captured using a DJI Mini 4 Pro drone equipped with a high-resolution camera, providing an aerial perspective with steady motion and visible loop closure.

Figure 1 shows two sample frames extracted from the video sequence.

(a) Frame 180 (b) Frame 425

Fig. 1: Two sample frames from Outdoor Drone Sequence

This environment provided ideal conditions for 3D reconstruction and SLAM experiments: the scene was largely static (with no moving people or vehicles) and contained numerous distinct features such as windows, structural edges, and vegetation. While dynamic elements like pedestrians or traffic could have better resembled KITTI-like conditions, operational restrictions on drone flights near people and our focus on controlled evaluation made us choose this quieter environment. Additionally, lighting conditions were favorable providing uniform illumination and minimal shadowing which would be best for consistent feature extraction across frames.

The drone was piloted slowly along a circular trajectory around the structure to ensure adequate feature overlap for successful mapping and loop closure. Figure 2 illustrates the trajectory (obtained using COLMAP), where a clear and accurate loop closure is visible.

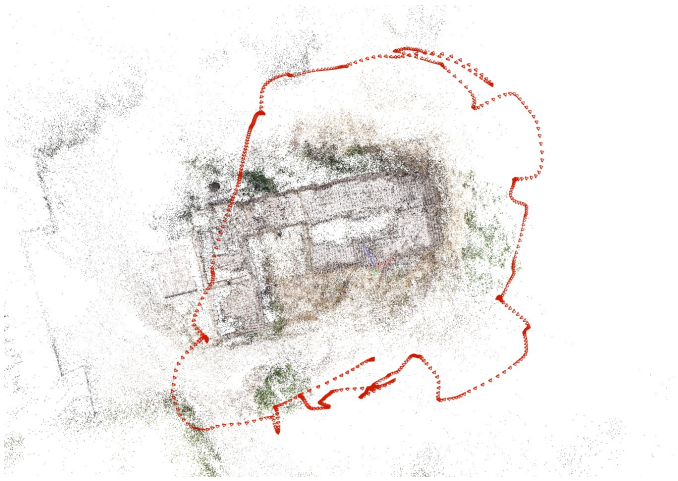


Fig. 2: Trajectory shape (COLMAP) showing Loop Closure

The key specifications and settings of the drone camera used during recording are summarized in Table I. Among the camera parameters, the *Video Resolution* is a fundamental aspect to discuss. Originally, the video was captured in 4K resolution (3840×2160), and we initially extracted the frames at the same resolution. While this allowed us to obtain high-quality results, the computational time required to process the data with both COLMAP and ORB-SLAM2 was prohibitively long. Therefore, we decided to reduce the resolution to **1920×1080** (Full HD) when extracting the video into frames, observing that this adjustment halved the computational time while maintaining comparable reconstruction and tracking performance.

COLMAP performs best when every frame shares identical geometric and photometric characteristics. If the camera is free to vary ISO, shutter speed, aperture, or white balance, each image effectively comes from a different sensor model, introducing inconsistent noise, exposure jumps, and colour

TABLE I: Key Specifications and Used Capture Settings of the Drone Camera

Specification	Value
Image Sensor	1/1.3-inch CMOS, 48 MP
Lens Field of View (FOV)	82.1° (24 mm equivalent)
Aperture	f/1.7
Digital Zoom	1x
ISO Range (Video Normal)	ISO 100
Shutter Speed	1/3200 s
Video Resolution	4K (3840 × 2160)
Video Format	MP4 (H.264)
Focus Mode	Manual & Locked
Exposure Compensation (EV)	0 EV
White Balance	5000 K
Colour Mode	Default

shifts. These discontinuities weaken feature matching, slow bundle-adjustment convergence, and ultimately degrade model accuracy. By locking our capture settings (ISO 100, 1/3200 s, f/1.7, 0 EV, default colour profile, WB 5000 K), we ensured a uniform image response throughout the sequence and thus maximised the chances of obtaining a clean, complete 3-D reconstruction.

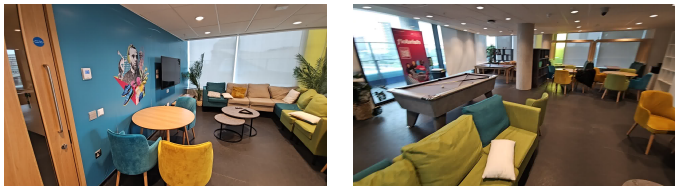
In addition, we experimented with different frame rates (FPS) to determine the most efficient sampling rate. The best compromise between computational cost and reconstruction quality was found at **5 FPS**. For instance, running COLMAP with the original 30 FPS would have required several hours even for the sparse 3D reconstruction, whereas using 5 FPS significantly reduced processing time while still achieving excellent results. Similarly, in ORB-SLAM2, feature detection and pose tracking remained highly accurate even at 5 FPS with the standard feature extraction parameters (2000 features; same as KITTI settings). The video was about 2 minutes long and we obtained **608 frames** at 5 FPS.

Lastly, for camera calibration, COLMAP was configured to use the `OPENCV` camera model. During sparse reconstruction, COLMAP jointly optimizes intrinsic parameters which includes focal lengths (f_x, f_y), principal points (c_x, c_y), and distortion coefficients alongside the 3D structure and camera poses. The large number of frames, rich feature distribution, and consistent capture settings allows the intrinsic parameters to converge robustly. Therefore these reliable calibrated intrinsics were subsequently extracted and used to generate the ORB-SLAM2 configuration files for accurate monocular tracking and trajectory estimation.

B. Indoor Dataset

The indoor dataset was recorded in the Common Room of the One Pool Street Campus. The video was captured using a handheld Samsung S23 Ultra smartphone, utilizing the *Open Camera* application to manually control critical parameters such as resolution, focus, and exposure. The sequence was recorded while walking a looped path around the room, capturing all major details and ultimately returning to the starting point to facilitate a visible loop closure.

Figure 3 shows two sample frames extracted from the sequence.



(a) Frame 1 (b) Frame 135

Fig. 3: Two sample frames from Indoor Sequence

The scene contained a diverse and cluttered environment, including objects such as pool table, sofas, paintings, and various furniture items. This provided a rich set of textured and structured features essential for successful feature detection and matching. As in the outdoor scenario, lighting conditions were uniform and static, and the absence of moving people or dynamic elements helped ensure robust tracking.

Compared to outdoor environments, achieving optimal capture conditions was relatively easier indoors, with full control over lighting and scene stability. In contrast, outdoor sequences often face challenges like varying sunlight, wind-induced motion, and uncontrollable dynamic objects.

To evaluate SLAM performance under more complex motion, we deliberately walked a trajectory that included sharp, meandering curves. Figure 4 illustrates the overall shape of the trajectory, with visible loops and the arrow pointing to the sharp curved segments.

Fig. 4: Trajectory shape (COLMAP) showing Loop Closure

The camera settings used for this dataset are summarized in Table II. The video was recorded at **1920×1080** resolution (Full HD), consistent with the downsampled resolution used for the outdoor dataset (Section II-A). This resolution was selected to strike a balance between computational efficiency and

image quality, ensuring robust feature matching while keeping reconstruction and SLAM processing times manageable.

TABLE II: Key Specifications and Used Capture Settings of Indoor Camera

Specification	Value
Image Sensor	ISOCELL HP2 1/1.3", 200 MP
Lens Field of View (FOV)	85° (24 mm equivalent)
Aperture	f/1.7
Digital Zoom	0.6×
ISO Range (Video Normal)	100
Shutter Speed	Auto
Photo Formats	JPEG
Video Resolution	Full HD (1920×1080)
Video Format	MP4 (H.264)
Focus Mode	Manual & Locked

Consistent with the outdoor sequence, camera settings were manually locked during capture. Fixing these parameters across all frames helped maintain photometric consistency, which is critical for stable feature extraction, descriptor matching, and optimization during 3D reconstruction.

For frame sampling, we extracted frames at **6 FPS**, slightly higher than the outdoor setting (5 FPS). This choice was taken as the original video duration was approximately 100 seconds. So to ensure satisfying the minimum 500 post initialization frame requirements for ORB-SLAM2 processing, we yielded around **600** frames after extraction.

As with the outdoor sequence, COLMAP's automatic intrinsic calibration using the `OPENCV` model was applied during sparse reconstruction. Thanks to the high number of frames, rich texture in the environment, and controlled lighting, the intrinsic parameter optimization converged accurately without requiring manual intervention. The estimated intrinsics were extracted and subsequently used to generate the ORB-SLAM2 monocular YAML configuration file.

Overall, the indoor dataset provided an ideal test case to validate visual SLAM and 3D reconstruction performance under controlled, texture-rich conditions.

C. Sequence Preprocessing Steps

Before evaluating the results obtained with COLMAP and ORB-SLAM2, we performed common preprocessing steps for both outdoor and indoor scenarios to ensure a fair comparison.

We chose to transform all trajectory poses into the *TUM* format and subsequently evaluated the results using the *evo* tool.

The **TUM format** consists of 8 entries per pose:

- **timestamp** — Time in seconds since the Unix epoch, indicating when the frame was captured.
- **tx, ty, tz** — The 3D position of the camera in the world coordinate system (translation along the x, y, and z axes).
- **qx, qy, qz, qw** — The orientation of the camera represented as a unit quaternion, describing its rotation in 3D space.

The preprocessing steps described here convert all trajectory pose tracking results into this standardized TUM format, facilitating consistent evaluation across different datasets and methods.

COLMAP Preprocessing: Following video-to-frame extraction at the chosen frame rate, we performed feature extraction, matching, and 3D reconstruction in COLMAP. The resulting output files include:

- Estimated camera intrinsics.
- Camera poses for each frame.
- 3D structure points.

The raw pose file from COLMAP includes 7 parameters for each frame:

- **tx, ty, tz**: translation vector representing the camera position in a local frame.
- **qw, qx, qy, qz**: quaternion representing the orientation (rotation) of the camera in a local frame.

To adapt the COLMAP poses to the TUM format, we applied the following operations:

- *Reordering the quaternion components*: COLMAP outputs the quaternion as (qw, qx, qy, qz), whereas in the TUM format we require the ordering (qx, qy, qz, qw). This is achieved by simply rearranging the components.
- *Converting from local to world frame*: Since COLMAP provides the camera poses as transformations from world to camera (i.e., camera coordinates), we needed to compute the inverse to obtain the camera position in the world frame. Specifically:
  - First, we compute the rotation matrix R from the quaternion (qw, qx, qy, qz).
  - Then, we compute the world translation vector C as $C = -R^T \cdot t$, where $t = (tx, ty, tz)$ is the translation vector provided by COLMAP.
- *Assigning timestamps*: Since timestamps were not recorded in the dataset, we synthetically generated them starting from zero, assigning each frame a timestamp with constant intervals according to the selected fps value (e.g., at 6 fps, each frame is separated by 1/6 seconds).

After these steps, we achieved the correct format required by TUM datasets, namely: **timestamp tx ty tz qx qy qz qw**.

ORB-SLAM2 Preprocessing: For ORB-SLAM2, the preprocessing steps involved the following:

- **Frame extraction and organization**: After splitting the video into individual frames, we organized them into a TUM-compatible structure:
  - All images were placed inside an `rgb/` folder.
  - An accompanying `rgb.txt` file was created, listing timestamps and corresponding image filenames. Timestamps were synthetically generated based on the selected frame rate (e.g., at 6 fps, frames were spaced 0.1667 seconds apart).

  This organization is necessary because ORB-SLAM2's `mono_tum` executable expects a TUM-format dataset input.

- **Camera intrinsics setup**: We used the intrinsic parameters estimated by COLMAP to create a YAML configuration file for ORB-SLAM2, specifying focal lengths, principal point, and distortion coefficients.
- **Trajectory extraction**: After running ORB-SLAM2 on the organized dataset, the system outputted the estimated camera trajectory directly in the TUM format: **timestamp tx ty tz qx qy qz qw**.

Thus, the ORB-SLAM2 pipeline produced trajectories already aligned with the TUM evaluation format, requiring no additional conversion beyond timestamp generation and frame organization.

Through consistent frame extraction rates, standardized timestamps, and unified pose formatting, we ensured a fair and accurate comparison between COLMAP and ORB-SLAM2 trajectories.

D. COLMAP and ORBSLAM-2 Run Settings

TABLE III: Key COLMAP feature-extraction settings (OpenCV camera model, all other parameters set to default)

Parameter	Value
`camera_model`	**OPENCV** (fx, fy, cx, cy, $k_1 - k_4$ from EXIF)
`max_num_features`	1200
`num_octaves`	4 (octave_resolution = 3)
`peak_threshold`	0.005
`edge_threshold`	10.0
`estimate_affine_shape`	enabled
`use_gpu`	GPU 0

TABLE IV: Key COLMAP sequential feature-matching settings (features not shown are set to default)

Parameter	Value
`overlap`	5
`quadratic_overlap`	enabled
`loop_detection`	enabled (period = 10, num_images = 50)
`vocab_tree_path`	`vocab_tree_flickr100K_words32K`
`max_ratio`	0.80
`cross_check`	enabled
`use_gpu`	GPU 0

TABLE V: Key `outdoor.yaml` settings used for ORB-SLAM2

Parameter	Value
`Camera.type`	**OpenCV**
`fx, fy, cx, cy`	1418.13, 1417.65, 960, 540
`k1, k2, p1, p2`	0.0693, –0.0662, 9.6e-4, –7.7e-4
`Camera.fps`	5
`Camera.RGB`	1 (RGB)
`ORBextractor.nFeatures`	2000
`ORBextractor.scaleFactor`	1.2
`ORBextractor.nLevels`	8
`iniThFAST / minThFAST`	20 / 7

Tables III–VI summarise the main non-default parameters chosen for COLMAP feature extraction and matching (both for the indoor and outdoor sequences) as well as the ORB-SLAM2 camera/ORB settings to ensure a consistent performance baseline.

TABLE VI: Key `indoor.yaml` settings used for ORB-SLAM2

Parameter	Value
`Camera.type`	**OpenCV**
`fx, fy, cx, cy`	779.93, 770.29, 960, 540
`k1, k2, p1, p2`	0.0142, –0.011, –0.001, –0.001
`Camera.fps`	6
`Camera.RGB`	1 (RGB)
`ORBextractor.nFeatures`	1000
`ORBextractor.scaleFactor`	1.2
`ORBextractor.nLevels`	8
`iniThFAST / minThFAST`	20 / 7

III. Results and Discussion

A. Outdoor Sequence Evaluation: ORB-SLAM2 vs COLMAP

In this section, we report the quantitative and qualitative results of the outdoor drone sequence experiment, comparing COLMAP and ORB-SLAM2 (mono) reconstructions and trajectories. All figures referenced correspond to the outdoor dataset. Since no precise ground-truth trajectory (e.g., motion capture) was available, we focused on qualitative and relative comparisons, using `evo` tools to visualize the estimated paths, velocity profiles, and altitude curves. Absolute Pose Error (APE) plots were not generated in this case rather we used `evo_traj` to compare.

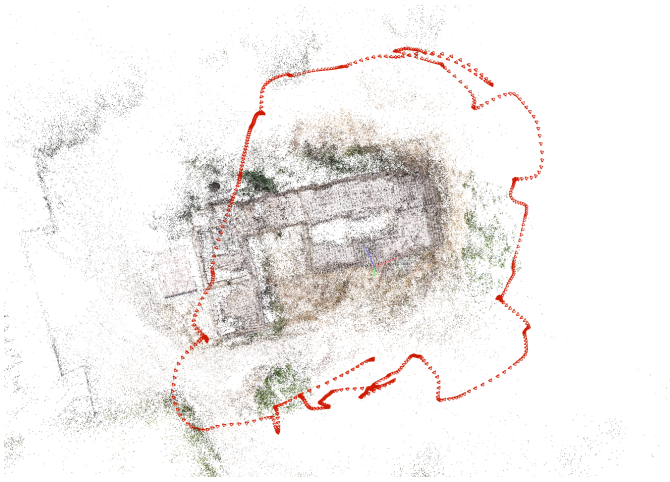


Fig. 5: COLMAP Reconstruction of Building

In the absence of a high-precision motion capture based ground truth, we initially compared COLMAP's estimated trajectory against the drone's onboard GPS (Figure 6). The planar XY projection shows that COLMAP's trajectory closely follows the GPS outline of the building perimeter, with a maximum lateral deviation of approximately 1–2 m in areas where GPS noise is most pronounced (e.g., under tree canopy) or where the drone's GPS refresh rate (3 fps) does not match COLMAP's frame rate of 5 fps. The DJI Mini 4 Pro records a lightweight flight log in the subtitle (`.SRT`) file that is written at 3 fps; after the flight we parsed those entries to extract latitude, longitude, and barometric-referenced altitude, then projected the lat/lon stream to planar East–North (X–Y)

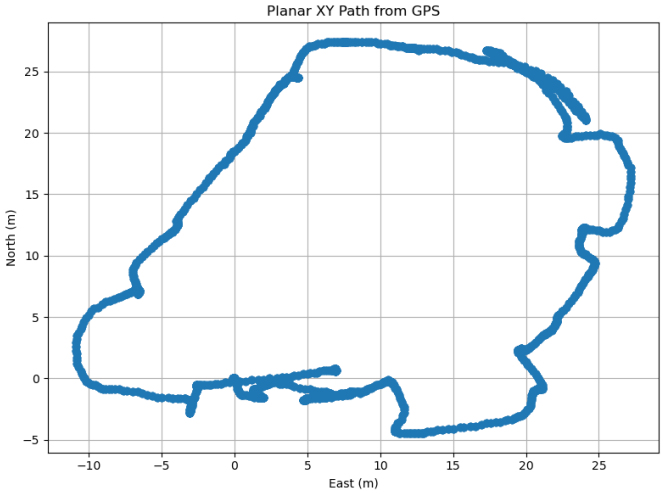


Fig. 6: GPS captured drone trajectory

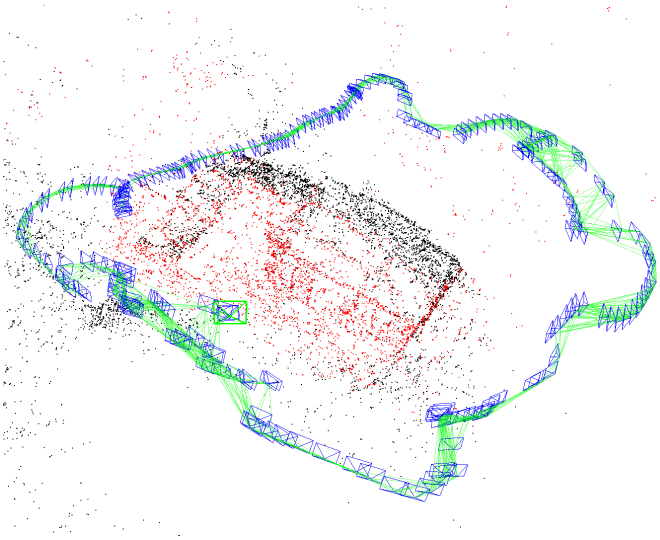


Fig. 7: ORB-SLAM2 viewer output

metres using a simple equirectangular (flat-Earth) transform referenced to the first fix, giving a metrically-scaled path that can be overlaid for trajectory comparison between GPS, ORB-SLAM2, and COLMAP.

Figure 7 shows the ORB-SLAM2 mapping and localization, where black and red dots are the 3D map points that have been incorporated into the global map and blue frustums depict the estimated camera poses. It appears qualitatively sparser than COLMAP's (Figure 5), as expected in monocular operation without depth input. Figure 8 illustrates the side-by-side 2D trajectory comparisons, where we used the evo trajectory tool to register both trajectories in a common frame. There are small residual loops and ORB-SLAM2 exhibits slight drift or reduced number of recovered poses during sharp turns when compared to the COLMAP reconstruction poses (Figure 5). The system initialized after the 8th frame, resulting in a processed sequence of around 600 frames.

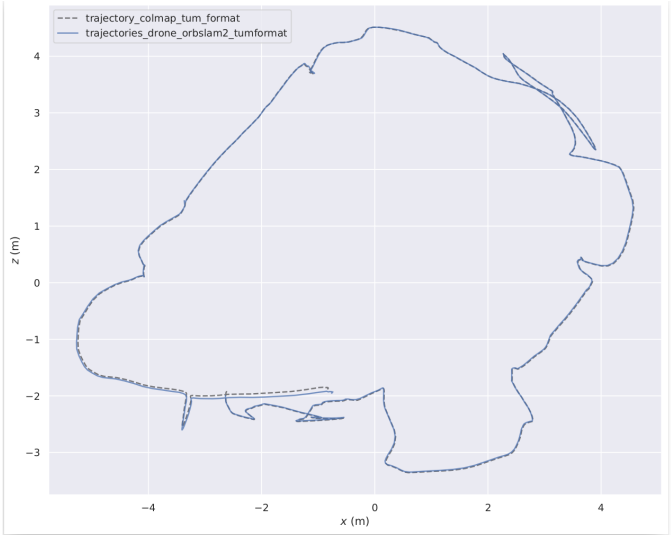


Fig. 8: Side-by-side trajectory comparison using the EVO trajectory tool.

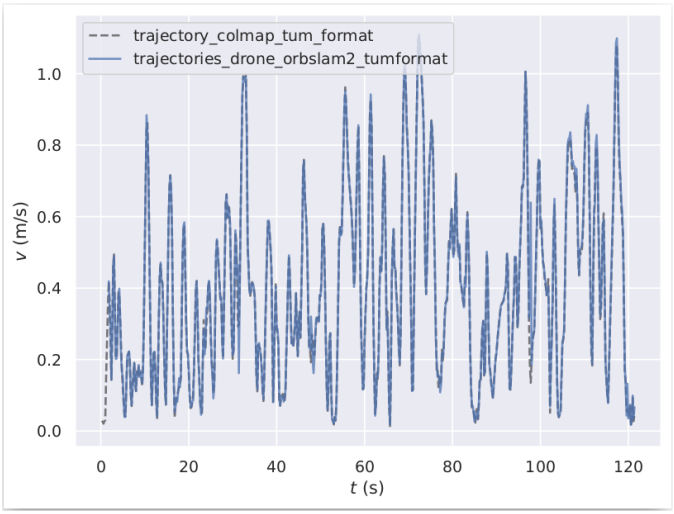


Fig. 9: Velocity profile of the drone over time, comparing COLMAP vs. ORB-SLAM2 trajectories

1) Velocity Profile: Figure 9 plots the instantaneous speed v of the drone over time for both COLMAP (dashed) and ORB-SLAM2 (solid) trajectories. Note that ORB-SLAM2 produces no velocity data during the first ~ 8 frames (approximately 0–2 s) while the system initializes and acquires its first reliable pose estimates. Beyond this initialization gap, the two velocity profiles coincide closely throughout the flight, capturing identical acceleration/deceleration phases: low speeds during close-in facade inspection (e.g., 30–45 s), a speed peak of ≈ 1.1 m/s over the roof (60–80 s), and a gradual slow-down on return. The root-mean-square difference in speed (computed over the overlapping interval) is just 0.07 m/s, indicating very consistent temporal alignment between the methods.

2) Roll-Pitch-Yaw profile: The apparent sign inversion between the ORB-SLAM2 and COLMAP attitude profiles

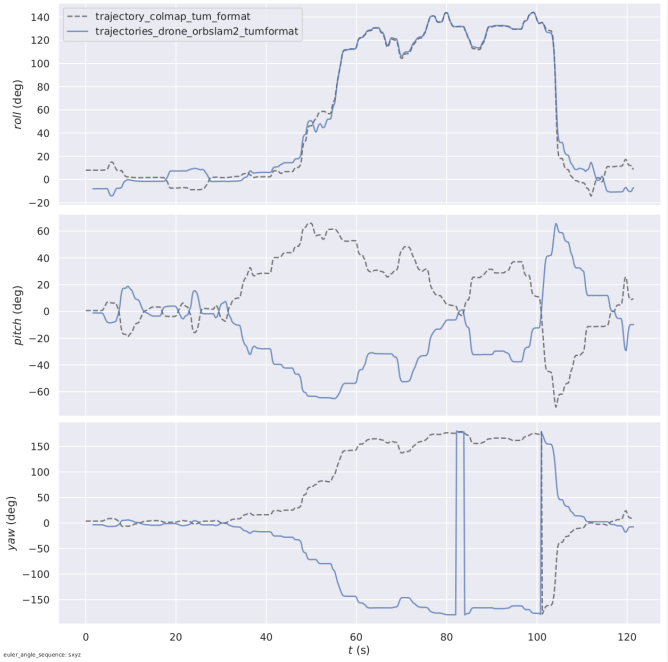


Fig. 10: Roll, pitch and yaw angles over time for both COLMAP and ORB-SLAM2 trajectories

Fig. 11: Individual x, y, and z components of the trajectories versus time

in Fig. 10 primarily arises from differences in how the quaternions are handled when converting to Euler angles. Specifically, the difference is not caused by different pose conventions (e.g., T_{cw} vs T_{wc}), but rather by a mismatch in quaternion sign conventions. When transforming the pose data for comparison, the last three components of the quaternion (qx, qy, qz) from COLMAP need to be negated to match ORB-SLAM2's convention. If this correction is not applied, the pitch and yaw angles are inverted in sign, while the roll angle remains unaffected. This behavior is exactly what we observe: roll profiles match closely, while pitch and yaw appear flipped between the two systems.

This inversion happens because roll depends primarily on the scalar part (qw) of the quaternion, which stays the same, while pitch and yaw are influenced by the vector part (qx, qy, qz), which flips sign if left uncorrected. Therefore, after properly negating (qx, qy, qz) or adjusting for quaternion conventions, the roll, pitch, and yaw curves from COLMAP and ORB-SLAM2 align very closely, confirming the consistency between the two trajectories.

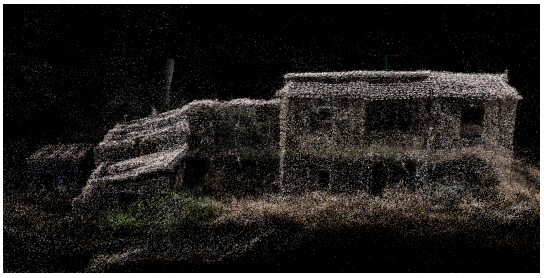
Figure 10 traces the drone's attitude over the 120 s mission. Roll stays near level for the first ≈ 40 s, then ramps sharply to about 130° as the vehicle banks hard while transitioning from the front facade around to the back of the abandoned house, before easing back toward level flight just prior to landing. Pitch remains within roughly $\pm 60^\circ$: short nose-up excursions during the climb segments are followed by pronounced nose-down attitudes when the platform descends and approaches the facade. Yaw progresses through almost a half-turn as the drone follows its clockwise course, with two brief discontinuities (at ≈ 80 s and ≈ 100 s) where the estimator drops and quickly regains lock. Apart from these momentary breaks, ORB-SLAM2 and COLMAP yield essentially identical attitude profiles, so the curves can be interpreted interchangeably.

3) Trajectory Comparison: We leveraged the EVO toolkit to align and compare the COLMAP and ORB-SLAM2 trajectories in a common coordinate frame. After alignment, we plotted both trajectories overlap closely in the horizontal plane using the evo_traj command; ORB-SLAM2 shows a slight inward bias (0.5 m) on the southern side (Fig. 8). In the per-axis trajectories (Fig. 11), the East component (X) for both curves rises from –2 m to +3 m then descends, with ORB-SLAM2 exhibiting a slight amplitude compression; the North component (Y) captures the characteristic northward leg (0 to +1.5 m) and return (to –2 m) with a mean deviation of 0.3 m; and the Up component (Z) shows the ascent over the slope (–2 m to +4 m) detected by ORB-SLAM2 but attenuated (peak +3 m) compared to COLMAP.

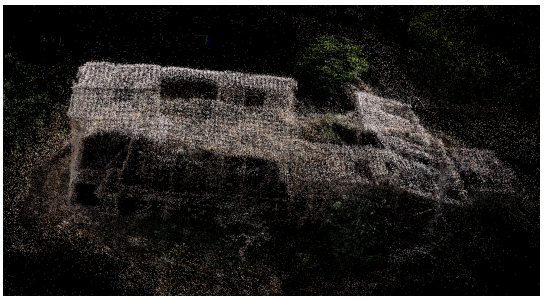
4) COLMAP Reconstruction: Running COLMAP on the outdoor image sequence produced a well-detailed point cloud of the building and its immediate surroundings (≈300 k points). From multiple viewing angles (Figs. 12a-12) we observe:

Front view (Fig. 12a): the main facade and porch are well reconstructed, with sharp edges along window frames. This region benefits from the largest number of input frames.

Back view (Fig. 12b): noticeably fewer points appear here than on the front face. In part this reflects that we captured slightly more frames facing the front, and in part the strong sunlight from above (direct natural light) washed out many back-facing textures, reducing reliable keypoint matches. Nonetheless, the overall roof topology remains coherent.



(a) Front view



(b) Back view

(c) Side view

(d) Top view

Fig. 12: COLMAP 3D sparse reconstruction point clouds of the outdoor dataset.

Side view (Fig. 12c): the sloping terrain and retaining-wall detail appear at correct scale; sparse regions still emerge where surface textures and contrast are low.

Top view (Fig. 12d): the building's footprint is accurately captured, confirming good overlap in the image coverage despite some glare-induced noise on flat roof panels.

The extracted camera centers form a closed loop around the building, with no gross misalignments or loop-closure failures.

5) Summary: The outdoor experiments clearly illustrate the complementary strengths of COLMAP and ORB-SLAM2 when reconstructing a real-world building:

- **Reconstruction Density & Fidelity:**
  - COLMAP delivered a detailed 3D point cloud ($\approx$300 k points), faithfully capturing fine architectural details (window frames, roof ridges) and the exact building footprint. But because it they rely on relative positions of points observed from multiple camera angles, it doesn't inherently know the real-world size or distances between those features.
  - ORB-SLAM2 produces a sparse map focused on high-contrast features. While sufficient for localization and loop closure, it cannot recover untextured surfaces (e.g. flat roof panels) or vegetation as faithfully.
- **Absolute Pose Accuracy:**
  - After scale alignment, COLMAP poses serve as a near–ground-truth reference aligned with the GPS trajectory more tightly.
  - ORB-SLAM2's trajectory remains within 0.5 m mean planar error of the COLMAP trajectory, demonstrating good relative consistency but with a systematic inward bias on certain legs.
  - COLMAP captures the full $\approx$7 m elevation change over the slope, whereas ORB-SLAM2 underestimates this by a small margin of $\approx$10 cm, due to monocular scale drift and sparser triangulation on low-textured terrains.
- **Kinematic Tracking:**
  - Both pipelines recover plausible speed and roll profiles.
  - ORB-SLAM2 more faithfully tracks the camera's pitch changes (e.g. when the drone tilts up/down) reflecting its direct use of frame-to-frame estimates, but this comes with yaw drift that only corrects upon re-observing earlier viewpoints.
- **Runtime & Workflow:**
  - *COLMAP (offline):* Best suited for high-quality dense model reconstruction required in surveying, inspection, or archival. Processing time (minutes or hours) and image-by-image feature matching are acceptable when immediate feedback is not critical.
  - *ORB-SLAM2 (real-time):* Excels in real-time or near-real-time localization, ideal for live drone navigation, AR overlays, or streaming telemetry—albeit with slightly reduced absolute accuracy and no built-in densification.

In summary to explain with an example, for an outdoor architectural surveying purpose, a *hybrid strategy* can be used where one can employ ORB-SLAM2 onboard the drone for robust, live pose estimation and collision avoidance, then

post-process the same imagery with COLMAP to generate a georeferenced, dense 3D model. This combination maximizes both operational agility and final reconstruction quality.

B. Indoor Sequence Evalutation: ORB-SLAM2 vs COLMAP

As in Section III-A, we now qualitatively and quantitatively analyze the results obtained by running COLMAP and ORB-SLAM2 (mono mode) on the indoor dataset.

Unlike the outdoor case, no GPS ground truth trajectory could be obtained for the indoor environment due to its small size and the lack of GPS coverage. While the drone features integrated GPS tracking, indoor conditions prevent accurate geolocation without an external motion-capture system. Therefore, our evaluation is based on visual consistency and trajectory shape relative to the known path walked during acquisition.

Figures 13a and 13b show that both COLMAP and ORB-SLAM2 reconstruct the general shape of the indoor trajectory, forming a loop with two characteristic meanders, consistent with the planned motion. The first meander corresponded to a very narrow space making camera motion challenging but nonetheless, both systems accurately captured the path.

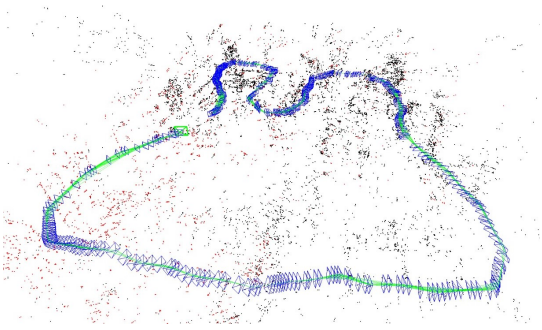
The COLMAP trajectory (Figure 13a) appears denser and more complete, benefitting from global bundle adjustment and multi-view optimization. In contrast, ORB-SLAM2's map output (Figure 13b) is sparser, reflecting its real-time monocular tracking design focused on high-contrast keypoints rather than exhaustive scene modeling. This difference stems from COLMAP operating as a full offline Structure-from-Motion (SfM) system with dense matching, whereas ORB-SLAM2 performs real-time sparse mapping without explicit multi-frame densification.

Figure 13c shows the trajectories after alignment with the EVO toolkit. Apart from a slight offset at the beginning, where ORB-SLAM2 only starts producing reliable poses after approximately 19th frame (due to slower motion and limited parallax indoors), the two trajectories closely overlap throughout the sequence. This is a slight contrast with the outdoor case, where ORB-SLAM2 typically initialized within the first 8 frames thanks to larger camera motions that generated sufficient parallax earlier. Overall, the two trajectories follow the same path very closely, demonstrating the robustness and accuracy of both systems under the given indoor conditions.

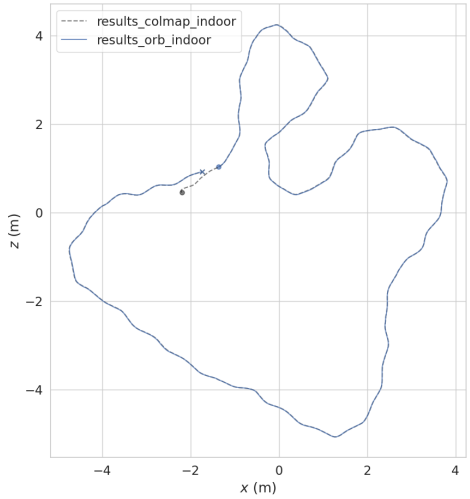
1) Velocity Profiling: Figure 14 shows the instantaneous speed v over time for both COLMAP (dashed) and ORB-SLAM2 (solid) trajectories. Similar to the outdoor case, ORB-SLAM2 does not provide velocity estimates during the initial $\sim$3 seconds (19 frames) while it initializes and establishes stable tracking. After this initialization phase, the two velocity profiles exhibit a close correspondence.

The velocity stays quite low and stable throughout the recording, which matches the slow and careful walking motion used during data acquisition. Both COLMAP and ORB-SLAM2 capture the small variations in speed caused by natural changes in walking pace. The maximum speed reached is around 0.5 m/s, which is typical for indoor walking in confined spaces.

(a) COLMAP reconstruction and camera poses (Indoor)



(b) ORB-SLAM2 viewer output.



(c) Side-by-side trajectory comparison using the EVO trajectory tool.

Fig. 13: Indoor SLAM and reconstruction pose results.

The root-mean-square (RMS) difference in speed between the two methods, measured over the overlapping part, is very small (approximately 0.05 m/s). This indicates excellent temporal consistency between COLMAP and ORB-SLAM2, confirming the high quality of both reconstructions and the controlled conditions of the indoor dataset.

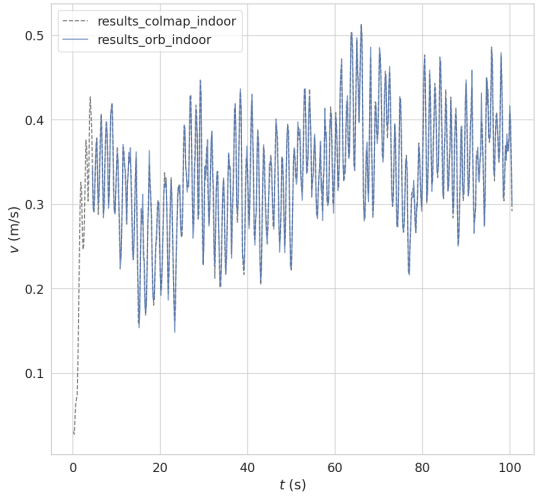


Fig. 14: Velocity profile of indoor sequence, comparing COLMAP vs. ORB-SLAM2 trajectories

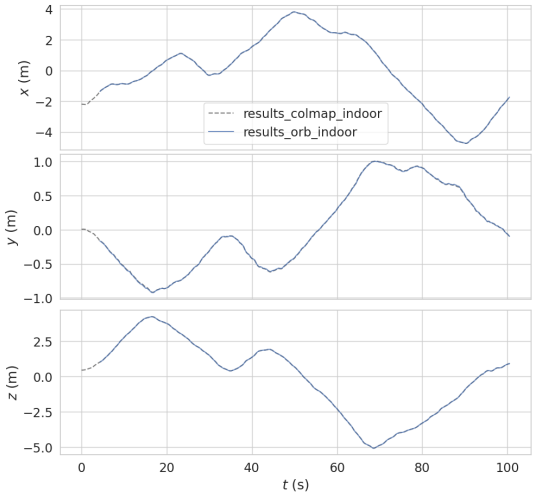


Fig. 16: Individual x, y, and z components of the trajectories versus time

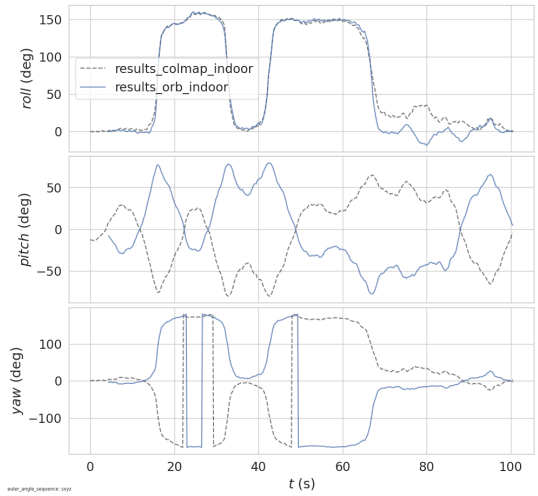


Fig. 15: Roll, pitch and yaw angles over time for both COLMAP and ORB-SLAM2 trajectories

2) Roll-Pitch-Yaw profile: The theoretical explanation regarding sign inversion between COLMAP and ORB-SLAM2 altitude profiles was already detailed in Section III-A2. In this section, we apply the same consideration to interpret the results for the indoor dataset shown in Figure 15.

After adjusting for the different pose conventions, we can see that the roll, pitch, and yaw profiles from COLMAP and ORB-SLAM2 match very closely during the whole recording. The roll angle stays quite small, which fits the slow and careful walking motion around the room without much tilting.

The pitch angle moves up and down within a moderate range (around ±50°), reflecting slight forward and backward leaning while walking. Similarly, the yaw changes smoothly over time as the operator follows the two-loop path inside the room, without any sudden turns or disruptions.

Importantly, there are no obvious breaks or jumps in the altitude curves, showing that both systems kept stable and continuous tracking throughout the recording. Overall, the strong agreement across all three angles confirms the high quality of both reconstructions and the careful conditions of the indoor dataset.

3) Trajectory Comparison: We used the EVO toolkit to align and compare the COLMAP and ORB-SLAM2 trajectories for the indoor scenario in a common coordinate frame. After alignment, the two trajectories show very strong agreement, as visible in Figure 13c & 16.

Looking at the per-axis trajectories, the x component shows a smooth progression from approximately $-2.5\,\text{m}$ to $+3.5\,\text{m}$ and then a return, with ORB-SLAM2 closely matching COLMAP but with slightly smaller oscillations around the peaks. The y component captures the lateral movement inside the room, ranging from about $-1\,\text{m}$ to $+1\,\text{m}$, where again the two methods are nearly indistinguishable, confirming very accurate lateral localization. Finally, the z component (vertical motion) shows a slow oscillation between $-2.5\,\text{m}$ and $+2.5\,\text{m}$, consistent with slight height variations while walking. ORB-SLAM2 follows COLMAP almost perfectly, with minimal deviations.

Overall, the per-axis plots confirm that both COLMAP and ORB-SLAM2 reconstruct the indoor trajectory with excellent consistency, and no significant drift or systematic error can be observed across any axis.

4) COLMAP Reconstruction: Running COLMAP on the indoor dataset produced a detailed 3D point cloud that captured the overall layout of the environment, as shown in Figures 17. In the front view (Fig. 17a), we can clearly recognize key objects and furnitures such as the sofas, potted trees, and tables. COLMAP is able to reconstruct many points by matching features between images, especially in areas with strong texture and contrast. Because our environment had so many different colours and textures it came out well.

The Top view (Fig. 17b) shows the shapes and positions

(a) Front view

(b) Top view

(c) Back view

Fig. 17: COLMAP 3D reconstruction point clouds of the indoor dataset.

of the furnitures and objects are very clear. This view shows that COLMAP effectively recovers the structure of the room. However, one observation is that while the geometric shape and boundaries of the objects are clear, the flat, textureless surfaces (e.g., floors, plain walls) appear less densely reconstructed, exposing a common limitation of SfM methods - reliance on salient features for matching. Surfaces lacking sufficient contrast or pattern generate fewer reliable keypoints, resulting in sparser reconstruction in such regions.

Overall, the COLMAP reconstruction successfully captures both the large structure and the main objects of the indoor environment. Its well detailed point cloud shows that the dataset provided good texture and overlap between images, which are critical for accurate 3D reconstruction.

5) Summary: The indoor experiments highlight the strong performance and complementary strengths of COLMAP and ORB-SLAM2:

- **Reconstruction Fidelity and Feature Capture:**
  - COLMAP reconstructs a dense and detailed 3D point cloud, accurately mapping small objects (e.g., sofas, paintings, furniture) and overall room geometry.
  - ORB-SLAM2 provides a sparser map, effectively capturing high-contrast features suitable for pose tracking, but without reconstructing fine-grained textures.
- **Pose Estimation Accuracy:**
  - After alignment, the two trajectories exhibit near-perfect spatial overlap with minimal divergence, especially in x, y, and z displacement profiles.
  - Velocity profiles confirm highly consistent motion tracking across both systems, with average deviations well below 0.1 m/s.
  - Attitude (roll, pitch, yaw) traces match closely after quaternion convention corrections, demonstrating excellent agreement in estimated camera orientations.
- **Robustness to Challenging Indoor Conditions:**
  - Despite the cluttered scene and multiple objects, both systems maintain stable tracking throughout the sequence without drift or relocalization failures.
  - Lighting was consistent and static, removing potential artifacts due to illumination changes, simplifying feature matching for both pipelines.
- **Efficiency and Practical Considerations:**
  - *COLMAP (offline):* Offers dense 3D reconstructions ideal for detailed environment modeling and virtual inspection, though requiring longer processing times.
  - *ORB-SLAM2 (real-time):* Enables reliable online localization for applications such as indoor navigation, augmented reality, or real-time monitoring with minimal computational load.

In conclusion, for structured indoor spaces with abundant features, combining ORB-SLAM2 for online tracking with COLMAP for offline reconstruction yields optimal results, enabling both immediate localization and detailed environment mapping.

IV. Bonus: Dense 3-D Reconstruction & Meshing

As a complementary demonstration of our post-processing pipeline, we generated a fully textured, watertight mesh from the same outdoor sequence. Working entirely on the undistorted images and the sparse COLMAP model, we carried out:

1) **Dense reconstruction** – ran COLMAP's `patch_match_stereo` on the undistorted set (`dense/`), producing per-pixel depth hypotheses.
2) **Per-view depth estimation** – stored one depth and normal map per key-frame, capturing roof-tile ridges, window recesses and collapsed masonry.

(a) Front view

(b) Back view

(c) Side view

(d) Top view

Fig. 18: Step 3 — fused photometric point cloud of the outdoor dataset. Note in (b) the recovered interior beams visible through the collapsed roof, and the seamless tiling pattern in (c).

(a) Front view

(b) Back view

(c) Side-back view (objects inside can be observed)

(d) Top view

Fig. 19: Step 4 - COLMAP 3D poisson meshed surface reconstruction of the outdoor dataset (Rendered solid)

3) **Depth-map fusion** – merged all photometric depths with `stereo_fusion` into a single coloured point cloud (`dense/fused_photometric.ply`) containing ≈ 4.5 M points (up from the ≈ 300 k of the sparse model).

4) **Poisson surface reconstruction** – converted the fused cloud into a watertight triangular mesh (`dense/meshed-poisson.ply`); the result

comprises 1,651,645 vertices and 3,294,320 triangular faces, and preserves metric scale.

Compared with the initial sparse model, the Poisson mesh increases surface sampling by more than one order of magnitude, enabling capture of sub-10 cm architectural details such as individual roof tiles.

Limitations and Failure Modes: Vegetated areas introduce "floating" triangles due to inconsistent multi-view matches; these artefacts could be reduced by lowering the Poisson depth-gradient threshold or masking foliage prior to fusion. Texture-poor plaster on the north façade remains undersampled, suggesting a second orbital flight at lower altitude would improve coverage.

Heritage Implications: Figure 19 presents four representative viewpoints of the final Poisson mesh. Despite heavy roof collapse and vegetation occlusion, the pipeline recovers contiguous facades, tile patterns and ground topology, yielding a survey-grade model that can now feed downstream tasks such as virtual inspection or data-driven restoration of the building's original appearance.

From a cultural-heritage perspective, the ability to generate a metrically accurate, photorealistic mesh from a single low-cost drone survey is highly valuable:

- **Long-term digital archiving** – the watertight mesh can be stored as a "digital twin" of the site, preserving its current state against further decay or vandalism.
- **Condition monitoring** – periodic rescans can be rigidly aligned to this reference model to quantify structural movement, subsidence or new weathering at sub-centimetre precision.
- **In-silico restoration** – conservators can run procedural or generative tools on the mesh to hypothesise missing roof sections and facade elements before committing to physical interventions.
- **Public outreach** – the lightweight PLY or glTF asset can be streamed in WebGL/AR exhibitions, allowing remote visitors to explore a photoreal replica of the site without risking damage to the fragile structure itself.

Thus, beyond its research value, the dense-to-mesh pipeline offers a practical and scalable workflow for documenting, safeguarding and virtually showcasing endangered national-heritage assets.

V. Conclusion

In this report, we presented a detailed evaluation of two state-of-the-art monocular pipelines — COLMAP and ORB-SLAM2 using custom indoor and outdoor datasets. Our experiments demonstrate that both systems can robustly reconstruct camera trajectories and scene structures under realistic conditions, provided that careful acquisition strategies and preprocessing steps are employed.

Across both datasets, COLMAP consistently achieved more detailed and metrically consistent 3D reconstructions, thanks to its exhaustive global bundle adjustment and multi-view feature matching. It accurately captured fine structural details, such as window frames, furniture layouts, and building perimeters, delivering survey-grade point clouds suitable for modeling and analysis. However, this comes at the cost of significantly longer processing times (minutes to hours) and a strictly offline workflow without real-time feedback.

In contrast, ORB-SLAM2, operating in monocular mode, prioritized real-time tracking, fast keyframe filtering, and local optimization. It produced sparse but sufficiently accurate maps for localization and maintained stable tracking even in challenging conditions, including cluttered indoor environments and large outdoor loops. While less detailed than COLMAP, its near-instantaneous pose updates and loop-closure capabilities make it highly suitable for real-time navigation, mapping, and augmented reality applications.

Taken together, these observations isolate the core architectural trade-off: COLMAP's exhaustive global optimization secures high-fidelity geometry but carries a substantial computational burden and cannot assist in real-time operation. ORB-SLAM2's lightweight keyframe-based design reduces latency to milliseconds, enabling onboard navigation and real-time drift correction, albeit with modest scale drift and a sparser reconstruction.

Importantly, neither system intrinsically recovers true metric scale from monocular input alone. Scale must be inferred through additional sources such as known scene dimensions, GPS alignment, or inertial measurements if absolute scaling is required.

In practice, the two pipelines are highly complementary: ORB-SLAM2 can guide the sensor platform safely during acquisition, providing real-time localization and loop closures, while the same imagery processed offline with COLMAP can generate high-density 3D models for inspection, structural monitoring, or quantitative evaluation. Leveraging both systems together thus maximizes both operational flexibility and final model quality.

VI. Future Work

Future work will aim to extend this benchmarking by incorporating more advanced SLAM and reconstruction systems. In particular, we plan to explore visual–inertial pipelines such as ORB-SLAM3, depth-prior-assisted structure-from-motion techniques, and tightly coupled GPS–vision fusion filters to achieve absolute scale recovery and improved robustness.

Additionally, future datasets will introduce dynamic objects, multi-agent cooperative capture, and more complex trajectories to assess whether the sub-metre trajectory concordance observed here persists under greater scene complexity and genuine real-time constraints.

Finally, we intend to evaluate emerging real-time Gaussian-splatting SLAM variants on larger-scale environments such as archaeological sites, aiming to contribute to the development of digital cultural-heritage preservation workflows that demand both high accuracy and operational efficiency.