UrbanSfM: Multi-View 3D Reconstruction and Photosynth-Style Virtual Navigation

Muhammad Musab Ali Chaudhry and Areesha Khan, *LUMS 2026*

***Abstract*—This project presents *UrbanSfM*, a complete educational Structure-from-Motion (SfM) system that reconstructs a sparse 3D representation of an indoor environment from a monocular image sequence and enables interactive exploration through a Photosynth-style virtual tour. The pipeline begins with SIFT feature extraction and pairwise matching, followed by the estimation of the essential matrix and recovery of the relative pose between an initial image pair. Using calibrated intrinsics, the baseline reconstruction is initialized by triangulating consistent correspondences. Subsequent views are registered incrementally via a PnP formulation that minimizes the reprojection error between observed 2D keypoints and existing 3D points, after which new points are triangulated and integrated into the map. Pose-only bundle adjustment is applied periodically to correct drift and maintain global consistency, producing a final reconstruction of 44,923 points.**

To enable intuitive visualization, a view-graph is constructed from camera connectivity, and an interactive WebGL viewer is implemented using Three.js. Camera transitions employ linear and spherical interpolation of poses, while image cross-fading enables smooth navigation between viewpoints. For demonstration purposes, a visually richer room-scale point cloud generated using Agisoft Metashape is incorporated to highlight the capabilities of the viewer. The full code, results, and viewer are publicly available in the project repository.

I. Introduction

Reconstructing three-dimensional structure from monocular image sequences is a central problem in computer vision, with applications in mapping, augmented reality, cultural heritage preservation, and robotics. Structure-from-Motion (SfM) methods recover both camera trajectories and sparse scene geometry by exploiting geometric constraints between overlapping views. Modern SfM pipelines typically combine feature detection, epipolar geometry, triangulation, and nonlinear optimization, forming a multi-stage estimation problem that must balance numerical stability, geometric correctness, and computational efficiency.

This project develops *UrbanSfM*, an end-to-end educational SfM system implemented over four weeks as part of the CS436 course on 3D Scene Reconstruction. The system begins with SIFT-based feature extraction and pairwise matching, followed by essential matrix estimation and initialization of a two-view reconstruction. The pipeline then extends to an incremental multi-view formulation in which new cameras are registered via the Perspective-*n*-Point (PnP) problem, and new 3D points are triangulated from multi-view correspondences. To ensure global consistency, pose-only bundle adjustment is performed periodically to minimize reprojection error and correct drifting trajectories.

Manuscript received Dec 7, 2025; revised Dec 8, 2025.

The reconstructed 3D map and estimated camera poses are further used to construct a view-graph that encodes image connectivity. This enables a Photosynth-style virtual navigation interface, implemented using Three.js, where pose interpolation and image cross-fading allow smooth transitions between viewpoints. For demonstration purposes, a visually richer room-scale point cloud generated using Agisoft Metashape is integrated into the viewer.

Overall, the project provides a complete pipeline from feature detection to interactive visualization, offering a pedagogically grounded exploration of geometric methods in multi-view reconstruction.

II. Methodology

This section describes the complete mathematical formulation and reconstruction logic of the system, structured around the classical geometry of multi-view vision. The pipeline consists of feature detection and matching, two-view initialization, incremental pose estimation, triangulation, point management, global refinement, and construction of a view-graph for the virtual tour. All algorithmic steps are expressed using standard multi-view geometry notation and avoid implementation-specific details.

A. Camera Projection Model

Each image is modeled by the pinhole camera projection equation

$$\mathbf{x} \sim \mathbf{K}\,[\mathbf{R} \mid \mathbf{t}]\,\mathbf{X}, \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^3$ is a world point in homogeneous form, $\mathbf{x} \in \mathbb{R}^2$ is its pixel measurement, $\mathbf{R} \in SO(3)$ and $\mathbf{t} \in \mathbb{R}^3$ describe the camera pose, and $\mathbf{K}$ is the intrinsic calibration matrix,

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

Normalized image coordinates are obtained by $\tilde{\mathbf{x}} = \mathbf{K}^{-1}\mathbf{x}$.

B. Feature Detection and Matching

For each image, distinctive local features are extracted and represented as descriptors. Given two images i and j, descriptor sets are matched using nearest-neighbor search, producing tentative correspondences $\{\mathbf{x}_k^{(i)} \leftrightarrow \mathbf{x}_k^{(j)}\}$. A ratio test filters ambiguous matches, leaving geometrically meaningful point pairs for epipolar estimation.

C. Relative Pose and the Essential Matrix

Given normalized correspondences $\tilde{\mathbf{x}}_k^{(i)}$ and $\tilde{\mathbf{x}}_k^{(j)}$, the essential matrix $\mathbf{E}$ satisfies

$$\tilde{\mathbf{x}}_k^{(j)\top}\,\mathbf{E}\,\tilde{\mathbf{x}}_k^{(i)} = 0. \tag{3}$$

To estimate $\mathbf{E}$, the normalized eight-point algorithm is used. For each correspondence, define the vector

$$\mathbf{a}_k = \begin{bmatrix} \tilde{x}_k^{(j)}\tilde{x}_k^{(i)} & \tilde{x}_k^{(j)}\tilde{y}_k^{(i)} & \tilde{x}_k^{(j)} & \tilde{y}_k^{(j)}\tilde{x}_k^{(i)} \\ \tilde{y}_k^{(j)}\tilde{y}_k^{(i)} & \tilde{y}_k^{(j)} & \tilde{x}_k^{(i)} & \tilde{y}_k^{(i)} \end{bmatrix} \quad 1 \Bigg]. \tag{4}$$

Stacking all rows yields a matrix $\mathbf{A}$ satisfying

$$\mathbf{A}\,\mathbf{e} = \mathbf{0}, \tag{5}$$

where $\mathbf{e}$ is the vectorized form of $\mathbf{E}$. Solving this homogeneous system via SVD produces an initial estimate $\mathbf{E}_0$, which is then projected onto the manifold of valid essential matrices by enforcing singular values $(\sigma, \sigma, 0)$:

$$\mathbf{E} = \mathbf{U}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}\mathbf{V}^\top. \tag{6}$$

The decomposition of $\mathbf{E}$ into rotation and translation follows the standard solution:

$$\mathbf{R}_1 = \mathbf{U}\mathbf{W}\mathbf{V}^\top, \quad \mathbf{R}_2 = \mathbf{U}\mathbf{W}^\top\mathbf{V}^\top, \quad \mathbf{t} = \mathbf{u}_3, \tag{7}$$

where $\mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $\mathbf{u}_3$ is the third column of $\mathbf{U}$. Four hypotheses arise; the correct one is selected by enforcing cheirality:

$$Z(\mathbf{X}_k) > 0 \quad \text{in both cameras.} \tag{8}$$

D. Initial Two-View Triangulation

Once the relative pose $(\mathbf{R}, \mathbf{t})$ is known, each correspondence $\mathbf{x}^{(i)}$ and $\mathbf{x}^{(j)}$ yields a linear triangulation constraint. For camera projection matrices

$$\mathbf{P}_i = \mathbf{K}[\mathbf{I} \mid \mathbf{0}], \qquad \mathbf{P}_j = \mathbf{K}[\mathbf{R} \mid \mathbf{t}], \tag{9}$$

a 3D point $\mathbf{X}$ must satisfy

$$\mathbf{x}^{(i)} \times (\mathbf{P}_i\mathbf{X}) = \mathbf{0}, \qquad \mathbf{x}^{(j)} \times (\mathbf{P}_j\mathbf{X}) = \mathbf{0}. \tag{10}$$

These yield four linear equations of the form

$$\mathbf{A}\mathbf{X} = \mathbf{0}. \tag{11}$$

Solving by SVD (taking the last column of $\mathbf{V}$) gives the homogeneous point $\mathbf{X}$. Depth positivity determines validity and outlier rejection.

E. Incremental Pose Estimation via PnP

For a new image k, existing 3D points visible in it provide 2D–3D correspondences $\{\mathbf{X}_j \leftrightarrow \mathbf{x}_j^{(k)}\}$. The camera pose $(\mathbf{R}_k, \mathbf{t}_k)$ is obtained by minimizing the reprojection error

$$\min_{\mathbf{R}_k, \mathbf{t}_k} \sum_j \left\| \mathbf{x}_j^{(k)} - \pi(\mathbf{R}_k\mathbf{X}_j + \mathbf{t}_k) \right\|^2, \tag{12}$$

where $\pi([\hat{x}, \hat{y}, \hat{z}]^\top) = \left(f_x \frac{\hat{x}}{\hat{z}} + c_x,\ f_y \frac{\hat{y}}{\hat{z}} + c_y \right)^\top$.

The rotation is parameterized using the minimal 3-vector Lie algebra representation,

$$\mathbf{R}_k = \exp([\boldsymbol{\omega}]_\times), \tag{13}$$

allowing Jacobians to be computed analytically.

Robust inliers are retained through a RANSAC scheme, improving accuracy in low-overlap frames.

F. Multi-View Triangulation

After estimating the new pose, additional points are triangulated between the new view and previously registered views. Given projection matrices $\mathbf{P}_i$ and $\mathbf{P}_k$, each valid correspondence produces a linear system

$$\mathbf{A} = \begin{bmatrix} x^{(i)}\mathbf{P}_i^{(3)\top} - \mathbf{P}_i^{(1)\top} \\ y^{(i)}\mathbf{P}_i^{(3)\top} - \mathbf{P}_i^{(2)\top} \\ x^{(k)}\mathbf{P}_k^{(3)\top} - \mathbf{P}_k^{(1)\top} \\ y^{(k)}\mathbf{P}_k^{(3)\top} - \mathbf{P}_k^{(2)\top} \end{bmatrix}, \qquad \mathbf{A}\mathbf{X} = \mathbf{0}, \tag{14}$$

whose solution again follows from SVD. This multi-view triangulation improves geometric stability and yields denser points.

G. Reprojection Error–Driven Pruning

Given N observations of a 3D point, the distribution of reprojection errors is analyzed:

$$e_j = \left\| \mathbf{x}_j - \pi(\mathbf{P}_{c(j)}\mathbf{X}) \right\|. \tag{15}$$

Points with errors exceeding a threshold relative to the median error are discarded, removing:

- mis-triangulated points,
- points from low-baseline configurations,
- points affected by incorrect correspondences.

This preserves geometric consistency before running global refinement.

H. Bundle Adjustment

Bundle adjustment refines all camera poses by minimizing the global reprojection error,

$$\min_{\{\mathbf{R}_i, \mathbf{t}_i\}} \sum_{(i,j)\in\mathcal{O}} \left\| \mathbf{x}_{ij} - \pi(\mathbf{R}_i\mathbf{X}_j + \mathbf{t}_i) \right\|^2, \tag{16}$$

where $\mathcal{O}$ is the set of all observed (camera, point) pairs. Since 3D points are fixed in our implementation, the optimization is over the 6-DoF parameters of each camera.

Linearization of the residuals yields the normal equations

$$(\mathbf{J}^\top\mathbf{J})\,\Delta = -\mathbf{J}^\top\mathbf{r}, \tag{17}$$

solved by Gauss–Newton with convergence monitored through cost reduction and optimality.

I. View-Graph Construction and Virtual Tour

After all poses are estimated, a view-graph is constructed where each node represents a camera and edges encode strong geometric overlap. Two cameras i and j are connected if:

- they observe a sufficiently large number of common 3D points,
- the relative baseline is adequate,
- and the corresponding images have consistent viewpoint transition.

For the interactive viewer, camera centers $\mathbf{C}_i = -\mathbf{R}_i^\top \mathbf{t}_i$ are visualized as nodes. Transitions between views interpolate camera trajectories smoothly:

$$\mathbf{p}(t) = (1-t)\,\mathbf{C}_i + t\,\mathbf{C}_j, \tag{18}$$

$$\mathbf{q}(t) = \mathrm{slerp}(\mathbf{q}_i, \mathbf{q}_j, t), \tag{19}$$

where $\mathbf{q}_i$ and $\mathbf{q}_j$ are the quaternion representations of rotations.

This produces realistic navigation through the reconstructed scene while remaining consistent with the underlying multi-view geometry.

III. Results

This section presents the quantitative and qualitative results of the full Structure-from-Motion pipeline. The evaluation focuses on (i) initialization quality, (ii) incremental reconstruction performance, (iii) behaviour of reprojection errors before and after bundle adjustment, and (iv) final map characteristics. Three visual results are shown: the Week 3 sparse point cloud produced by our pipeline, the higher-coverage Agisoft Metashape reconstruction used for the viewer demonstration, and the interactive virtual-tour interface.

A. Two-View Initialization

The reconstruction was initialized from image pair $(1, 2)$, which produced:

- 1454 feature matches,
- 700 inliers after essential matrix estimation,
- 700 successfully triangulated points,
- average parallax: 479.38 pixels.

The initial reprojection error was already low, indicating a numerically stable baseline. Table I summarizes the initialization accuracy.

TABLE I
Reprojection error during two-view initialization.

	Mean (px)	Median (px)	Max (px)
Before BA (global)	0.3456	0.3326	0.9415
After BA (global)	0.3373	0.3185	1.0590

The stability of the baseline becomes clear from the fact that the first-stage bundle adjustment required only a small refinement to converge.

B. Incremental Reconstruction Performance

As new images were registered, the number of 3D points grew steadily from 700 to $47{,}700$ before pruning, and $44{,}923$ after the final pruning stage. Table II summarizes the evolution of the 3D map.

TABLE II
Growth of the 3D point cloud during incremental SfM.

Stage	Points Added	Cumulative Points
Initialization (views 1–2)	700	700
After view 6	4087	4794
After view 10	6748	11542
After view 16	17161	28703
After view 23	19000	47700
Final (after pruning)	–	44923

Incremental PnP was generally stable; however, views with poor overlap or weak texture (e.g., views 8, 18, and 19) exhibited significantly fewer inliers (as low as 19) and required RANSAC to avoid catastrophic drift.

C. Reprojection Error Evolution

Reprojection error was the primary metric for geometric consistency. Each periodic bundle adjustment dramatically reduced both global error and per-image outliers.

a) After Adding View 9: The reconstruction reached $10{,}300$ points but accumulated substantial drift. Global mean reprojection error increased to 19.25 px before BA and dropped to 5.33 px afterward.

b) After Adding View 16: With $28{,}703$ points, the unrefined reconstruction showed a large error spike (mean 45.54 px, max $181{,}748$ px). After BA, the global mean fell to 8.89 px and the maximum error reduced to ~ 228 px.

c) After Adding View 23: The reconstruction reached its largest unrefined state ($47{,}700$ points) and exhibited severe drift (mean 106.10 px, max $699{,}120$ px). After pruning and BA, the map stabilized at $44{,}923$ points with:

$$\text{global mean} = 9.41 \text{ px}, \quad \text{median} = 5.21 \text{ px}, \quad \text{max} = 263.70 \text{ px}.$$

D. Point Pruning and Map Refinement

Three pruning stages removed outliers caused by:

- low baseline triangulation,
- mismatched features,
- weak geometric constraints (especially in textureless regions).

A total of 2777 points were removed across all pruning passes, producing a geometrically coherent final map.

TABLE III
Effect of pruning stages.

Stage	Points Removed	Points Remaining
After view 9	643	9657
After view 16	1899	26804
After view 23	395	44923

E. Qualitative Reconstruction Results

Figure 1 shows the final Week 3 sparse point cloud generated by the full incremental pipeline. Due to limited spatial coverage in the captured dataset, the reconstruction captures one portion of the environment with moderate geometric fidelity.

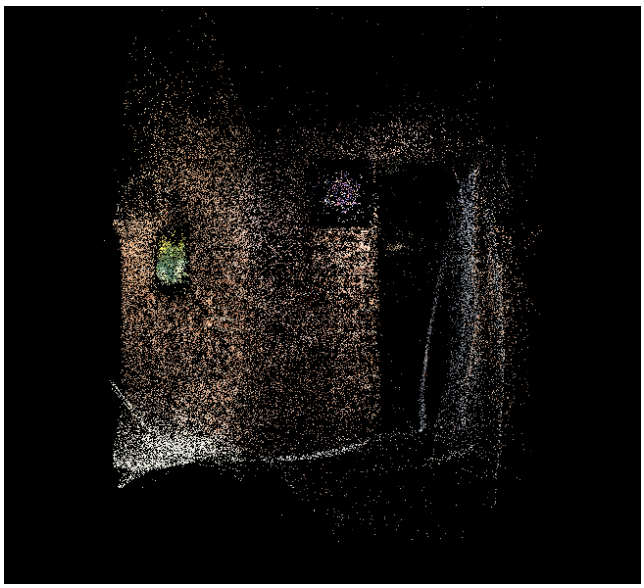

Fig. 1. Sparse 3D point cloud produced by our Week 3 SfM pipeline.

To demonstrate a more visually interpretable virtual tour, we reconstructed a complete room using Agisoft Metashape. Only a *sparse* version of this cloud is included in the repository due to size limitations.

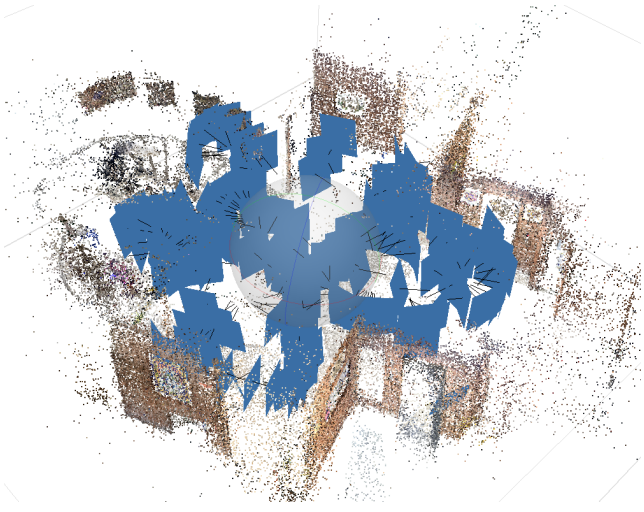

Fig. 2. Sparse Agisoft Metashape reconstruction used for the virtual-tour demonstration.

Finally, the interactive viewer was constructed using the camera poses and a view-graph computed from the Metashape dataset. Smooth viewpoint transitions were achieved using linear interpolation of camera centers and spherical interpolation of rotations.

Fig. 3. Screenshot of the pcd made in Agisoft Metashape.

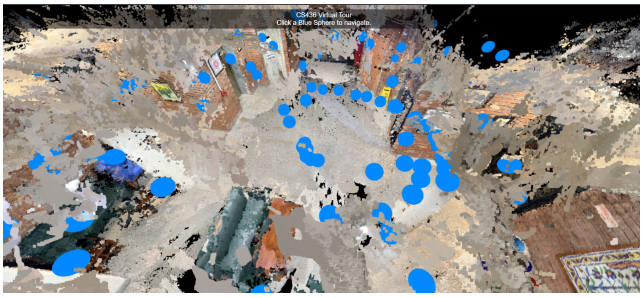

Fig. 4. Screenshot of the interactive virtual tour interface.

Overall, the results validate the correctness of the reconstruction pipeline, demonstrate the importance of bundle adjustment for stabilizing drift, and show how the recovered poses can be leveraged to build an interactive virtual navigation experience.

IV. Discussion

The experiments highlight both the strengths and the limitations of the implemented pipeline. This section analyzes why the algorithm behaves as observed, with particular attention to feature statistics, texture properties, bundle adjustment behaviour, pruning, and the virtual tour.

A. Texture, Feature Density, and Reconstruction Stability

A consistent empirical observation was that the pipeline behaved very differently across surface types:

- highly textured, structured surfaces such as patterned brick walls,
- extremely busy but weakly structured regions such as bedsheets,
- almost textureless regions such as plain painted walls.

The feature detector and descriptor operate on local image gradients. Patterned brick walls provide a rich distribution of corners and edges with well-localized, repeatable keypoints. In these regions, local image patches are distinctive enough that descriptors can be matched reliably across large baselines, leading to well-conditioned estimation of the essential matrix and robust triangulation. This explains why the reconstruction is most stable in wall segments with visible brick patterns.

Bedsheets, in contrast, produced an over-abundance of features without strong global structure. Many patches have similar high-frequency texture, so descriptors become locally ambiguous: multiple candidate matches may lie near the same position along the epipolar line. Even though the raw number of matches is high, the fraction of *geometrically consistent* correspondences is lower. In practice, this manifested as an "explosion" of the graph: a very dense feature graph with many weak or spurious edges, which in turn led to noisy or scattered triangulated points in those regions.

Plain walls represent the opposite failure mode: there are too few informative gradients to produce reliable keypoints. With insufficient correspondences, the epipolar geometry becomes ill-conditioned and the PnP step has very few 2D–3D correspondences to lock onto. The log segments where only a few dozen inliers were found (e.g., weak views such as 8, 18, and 19) are consistent with this interpretation: low-texture views yielded fragile pose estimates, which then propagated as large reprojection errors before bundle adjustment re-stabilized the solution.

Overall, the pipeline is best described as operating in a "sweet spot" of texture: structured, moderately rich patterns (brick, furniture edges, corners) are ideal; unstructured high-frequency patterns and large flat regions are problematic, albeit for opposite reasons.

B. Bundle Adjustment, Computational Cost, and the Role of Pruning

The reprojection error statistics show that periodic bundle adjustment is essential for maintaining geometric consistency. After adding view 9 and view 16, the global mean error rose to 19.25 px and 45.54 px respectively, with extremely large outliers (up to 10^5–10^6 pixels in the worst case). These spikes occur when incremental pose estimates and point triangulations accumulate drift, particularly in areas with weak baseline or poor texture.

Bundle adjustment solves a large coupled nonlinear least-squares problem over camera poses and 3D points. Its cost scales with the number of parameters, so running it on all points at every step would be prohibitively expensive. In this implementation, BA is triggered only after a fixed number of new views and is restricted to pose-only optimization, but the problem still involves tens of thousands of residuals.

Pruning was therefore introduced as a critical computational and geometric safeguard. By thresholding points based on their reprojection error (relative to the median), the system discards the most inconsistent 3D points before the next BA stage. This has two effects:

1) It removes gross outliers that would otherwise drag the optimizer into poor local minima or require more iterations to correct.
2) It reduces the dimensionality of the problem, making subsequent BA runs faster and numerically better conditioned.

The logs confirm this behaviour: after each pruning stage, the number of points drops (e.g., from $47{,}700$ to $44{,}923$ before final BA), while the global error stabilizes around a mean of ~ 9 px and median of ~ 5 px. The fact that the post-BA median is consistently low suggests that the residual error is dominated by a relatively small subset of challenging regions (e.g., weak-texture views), while the bulk of the reconstruction is internally consistent.

From a geometric standpoint, this justifies the design choice of combining aggressive pruning with pose-only BA: the algorithm is effectively trading completeness (keeping every triangulated point) for robustness and computational feasibility. The final map is sparser than the raw triangulation but significantly more stable.

C. Weak Views, Drift, and Global Consistency

The views flagged as weak (e.g., 8, 18, 19) illustrate how incremental SfM degrades when PnP receives only a small number of inliers. In such cases, the pose estimate is still constrained by the essential geometry, but the uncertainty in $[R \mid t]$ is much larger. When these uncertain poses are used as baselines for further triangulation, depth becomes poorly conditioned and the resulting 3D points may be scattered far from the true surface.

The extremely large pre-BA maxima (on the order of 10^5–10^6 pixels) are signatures of these numerical failures: a few 3D points are projected to locations far outside the image domain because their estimated depth is inconsistent with the true scene. Bundle adjustment, together with pruning, effectively "pulls" the solution back toward the configuration that best satisfies all multi-view constraints.

Importantly, the final error distribution is not uniform across images: some views converge to low mean errors (around 6–8 px), while the hardest views retain higher mean and median errors. This non-uniform pattern reflects the underlying image content and viewpoint geometry, and it emphasizes that even a globally consistent reconstruction may contain local pockets of higher uncertainty.

D. Virtual Tour Behaviour and Handling of Limited Overlap

The virtual tour is built on top of the recovered camera poses, but it does not assume perfect global consistency. Instead, it uses the view-graph structure and smooth interpolation in 3D to provide a perceptually coherent experience even when the metric reconstruction is imperfect.

Camera centres that share sufficient visual overlap are connected by edges in the view graph. When the user steps from one view to a neighbour, the viewer interpolates:

- the camera centre using linear interpolation in $\mathbb{R}^3$, and
- the orientation using spherical interpolation on $\mathrm{SO}(3)$.

This continuous interpolation acts as a low-pass filter on pose noise: small inconsistencies in the estimated rotations and translations are visually smoothed out along the transition trajectory, so the user perceives a stable navigation even if the underlying poses are only approximate.

In regions where the dataset has limited or no overlap (for example, two walls that were captured with no connecting views), the view graph simply has no edge. The viewer

therefore does not attempt to interpolate between these disconnected components; instead, the user can "jump" directly by selecting another camera node. The 3D point cloud remains visible as a global context, but the navigation logic respects the underlying connectivity implied by the data.

For the demonstration, an additional Agisoft Metashape reconstruction of a full room was used. This provided a denser and more spatially complete set of camera poses than the student-captured Week 3 dataset, which was limited to a smaller corner. However, the virtual tour logic is identical in both cases: only the quality and coverage of the underlying poses change. This separation between reconstruction and visualization is important: the viewer does not "fix" the geometry, but it exposes how far a given set of poses can support smooth navigation.

E. Why the Pipeline Works, and Where It Breaks

The overall behaviour of the system can be summarized as follows:

- **It works best when**: the scene provides structured texture, the baselines between views are moderate, and each new view has overlapping content with several previously reconstructed views. In this regime, epipolar geometry is well conditioned, triangulation is stable, and bundle adjustment mainly refines an already good solution.
- **It degrades when**: the scene contains large untextured regions, highly repetitive fine textures, or views with minimal overlap. In those cases, PnP receives few or ambiguous correspondences, triangulated points become unreliable, and reprojection errors spike until pruning and BA remove the offending structure.

The combination of incremental pose estimation, robust feature matching, periodic bundle adjustment, and reprojection-error-based pruning explains why the final map is usable despite these challenges. At the same time, the limitations observed in bedsheets and plain walls indicate clear directions for improvement, such as integrating multi-scale feature selection, regularizing depth in poorly constrained directions, or incorporating semantic or planar priors for man-made environments.

From a virtual-tour perspective, the viewer further demonstrates that a mathematically imperfect reconstruction can still support a convincing navigation experience, provided that the connectivity structure and relative poses are qualitatively correct. The Three.js interpolation and view-graph filtering therefore play a complementary role: they do not replace geometric accuracy, but they make the reconstructed scene explorable in a way that exposes both its strengths and its failure modes.

V. Conclusion

This work implemented a complete end-to-end Structure-from-Motion pipeline, progressing from feature extraction and two-view initialization to a full incremental reconstruction with periodic bundle adjustment, point pruning, and virtual-tour visualization. The experiments demonstrate that reliable reconstruction emerges when views exhibit sufficient texture, geometric structure, and overlap, while regions with repetitive or textureless surfaces introduce ambiguity, weak baselines, and drift. The combination of pose-only bundle adjustment and reprojection-error pruning proved essential for maintaining global consistency as the number of views increased. Although the Week 3 dataset produced a spatially limited reconstruction, the recovered camera poses were accurate enough to support a smooth and interpretable virtual navigation experience. A larger Metashape-generated model further illustrated how the same visualization framework generalizes to full-room environments. Overall, the results highlight both the strengths of incremental SfM and the practical challenges posed by real indoor scenes, pointing toward potential future improvements such as integrating multi-scale features, planar constraints, or denser nonlinear triangulation.