

Stable and Scalable Bundle Adjustment of Holistic 3D Structures

Shaohui Liu¹, Rémi Pautrat², Daniel Barath¹, Richard Hartley³, Viktor Larsson⁴, and Marc Pollefeys^{1,2}

¹ETH Zurich ²Microsoft Spatial AI Lab

³Australian National University ⁴Lund University

Abstract. Bundle Adjustment (BA) is a cornerstone of 3D computer vision and has benefited from decades of advances in sparse optimization and numerical methods. It was originally developed for jointly optimizing camera intrinsics, poses and sparse 3D points. While extensions incorporate lines and other primitives, integrating richer geometric structures such as parallelism, coplanarity, or wireframes often introduces significantly increased computational cost and reduced numerical stability. In this paper, we propose a unified framework that extends bundle adjustment to jointly optimize geometric features and higher-order relations. We first introduce a taxonomy that distinguishes scalable geometric *features* with direct 2D measurements (e.g., points and lines), from *groups* encoding higher-order relations (e.g., coplanarity, parallelism, etc.), where we show that groups can be modeled as camera-like entities within the bundle adjustment framework. Building on this formulation, we propose that both group constraints and cross-feature relations (i.e., point–line associations) can be expressed through 2D reprojection measurements. By formulating group-induced and cross-feature reprojection errors, we preserve the sparsity structure of classical point-based BA under Schur elimination, while avoiding direct 3D regularization that degrades the conditioning and stability. Experiments on both real-world and synthetic datasets demonstrate runtime performance comparable to classical point-only bundle adjustment, while producing significantly richer 3D structures and improved geometric accuracy.

1 Introduction

Bundle adjustment plays a central role in 3D reconstruction, jointly refining camera parameters and scene geometry by minimizing reprojection errors across all observations. Decades of research have produced highly efficient solvers that exploit the characteristic sparsity of the problem: because each 3D landmark couples only to its observing cameras, the Hessian over landmarks is block-diagonal, enabling Schur complement elimination that reduces the system to a compact camera-only problem. This structure has been the key to scaling bundle adjustment to large reconstructions with millions of points [2, 53, 73].

Yet real-world scenes, particularly man-made environments, contain geometric structures far richer than isolated point landmarks. Lines appear along

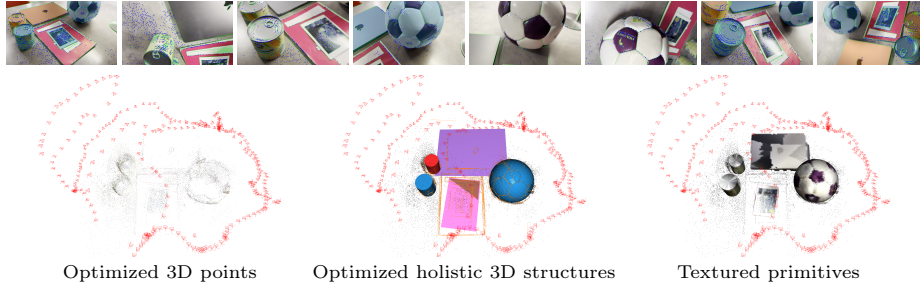


Fig. 1: Bundle adjustment with structural constraints. From multi-view images with detected points, lines, and groups (top), our proposed unified bundle adjustment framework is able to jointly optimize features (points/lines), camera poses, and structural constraints while preserving computational efficiency. This yields **richer optimizable sparse 3D reconstruction with geometric primitives** (planes, spheres, cylinders, etc.) beyond point clouds (left).

edges and boundaries, often in parallel configurations governed by vanishing points [51]. Surfaces form planes and other primitives that constrain the positions of points and lines alike. At finer scales, points and lines meet at junctions to form wireframe structures [27, 77, 85] that encode the geometric skeleton of buildings, furniture, and urban infrastructure. These structures provide powerful constraints that, if properly exploited, can improve reconstruction accuracy and completeness. However, incorporating them into bundle adjustment has remained a long-standing challenge due to two fundamental difficulties.

The first challenge is structural coupling. In classical bundle adjustment, the efficiency of the Schur complement relies on the fact that 3D landmarks are mutually independent: no residual couples two landmarks directly. Introducing geometric relations such as coplanarity, parallelism, orthogonality, or wireframe connectivity fundamentally violates this property. When two features share a geometric relation, their parameters become coupled, creating off-diagonal blocks in the landmark Hessian that destroy its block-diagonal structure. With many such associations, the efficient Schur elimination may break down entirely, leading to denser fill-in and increased computational cost.

The second challenge is the scale ambiguity coming with 3D constraints. Even when sparsity is handled, formulating structural constraints as direct 3D penalties (*e.g.*, point-to-plane distances [35, 58]) mixes 3D metric residuals with 2D pixel-space terms, degrading conditioning due to scale mismatch and introducing ad hoc weighting. Because such residuals lack a principled noise model, there is no natural way to anchor structural entities through well-observed features. Moreover, 3D penalties break the scale gauge ambiguity of bundle adjustment, biasing the optimizer toward shrinking the reconstruction to reduce the penalty.

In this work, we present a unified bundle adjustment framework that integrates geometric structures while addressing both challenges. We introduce a taxonomy on features and groups, which distinguishes geometric features with direct 2D measurements (points, lines) from groups encoding higher-order relations (vanishing points, planes, conics, etc.). This taxonomy provides a formu-

lation for incorporating diverse geometric entities under a single optimization scheme: virtual groups are added to the optimization problem and are placed alongside cameras in the non-eliminated block of the normal equations, absorbing the coupling between associated features and preserving the block-diagonal structure over the feature block.

Based on this formulation, we express both group and wireframe constraints as reprojection-based residuals in pixel space. Group-induced reprojection errors measure the pixel displacement from projecting a feature onto the group surface, while cross-feature reprojection errors use the observation of one feature to constrain another. This avoids direct 3D regularization and implicitly weights each constraint by the uncertainty (via Fisher information) of the involved features, naturally anchoring structural entities through more reliable associations.

We validate our framework through experiments on synthetic and real-world datasets. Our proposed framework is able to optimize on top of significantly richer reconstructions with holistic 3D structures (as in Fig. 1), while achieving runtime performance comparable to classical bundle adjustment and consistent improvements on camera poses and geometry. Code for the proposed bundle adjustment framework and the resulting SfM system are made available at <https://github.com/cvg/limap>, fully compatible with the COLMAP ecosystem.

2 Related Work

Advances in Bundle Adjustment. Bundle adjustment [63] has been widely studied and applied in 3D reconstruction for decades. Sparse matrix techniques and the Schur complement [25] have enabled efficient solutions by exploiting the bipartite structure between cameras and 3D points. There have been many efforts to improve its scalability via parallel/distributed computing [2, 20, 74] and advanced solvers [28, 41]. Other advances also include robust cost functions [80], incremental formulation [29], and differentiable BA for end-to-end learning [59, 60]. Building upon these advances, several open-sourced libraries have emerged and become standard tools, notably including g2o [32], Ceres solver [1], GTSAM [16], etc. These libraries work together with sparse linear algebra libraries [12, 15] and provide a nice ecosystem for modern developments of geometric 3D vision, enabling multiple reliable systems on structure-from-motion [44, 53, 57, 73] and SLAM [19, 31, 45] across years. A separate line of work reformulates the linear system, either by marginalizing landmarks with QR decomposition that never forms the Schur complement explicitly [17] or by expanding the matrix inverse as power series [71]. Recently, bundle adjustment is also proved to be effective in refining the poses and geometry of feedforward 3D models [11, 65–67, 70]. In this work, we introduce a unified framework extending bundle adjustment to handle holistic 3D structures with higher-level primitives and relations.

Geometric Primitives in Optimization. While bundle adjustment is traditionally formulated over 3D points, extending it to general geometric primitives dates back to the 1990s [26]. However, most recent works that incorporate object-

level primitives [33, 36, 46, 75, 78] directly optimize high-dimensional variables representing the primitives through 2D modeling of primitive observations. Such formulations typically ignore the structural relationships between primitives and their supporting features, which prevents seamless integration with classical bundle adjustment over 3D points.

Among geometric primitives, 3D lines have been extensively studied. In [5], Bartoli and Sturm formulate bundle adjustment over infinite 3D lines using an orthonormal parameterization of Plücker coordinates, which has since been adopted in many modern line-based optimization systems [39, 40, 76, 87]. These systems often further exploit structural cues such as vanishing points [51, 61], Manhattan-world assumptions [14] and their variants [34, 52], as well as learned monocular priors [30]. In practice, these systems can scale to thousands of lines, which motivates us to treat lines as feature-level entities similar to points.

Planes are another fundamental primitive, especially in man-made environments. Planar constraints have long been studied in geometric optimization, including formulations based on two-view homographies [21], auto-calibration [62], and explicit parameterization of 3D structures [6]. The use of homography constraints has recently been revisited in [83] for computational efficiency. However, this formulation requires enumerating over image pairs. Explicit parameterization of 3D structures [6] is conceptually appealing, but is sensitive to incorrect feature-plane associations, which are difficult to avoid in practice. Many recent works [9, 35, 56, 58] therefore introduce soft point-plane regularization terms in 3D, which have also been extended to line-plane relationships [4, 35]. However, unlike point-to-plane formulations in LiDAR or RGB-D SLAM [10, 23, 43, 84], which benefit from principled noise models on 3D measurements, monocular systems typically measure errors in 3D space at absolute scale with heuristic weighting. In contrast, our framework jointly optimizes multiple types of primitives using a unified residual formulation in pixel space, which implicitly models the uncertainty of different feature-primitive associations.

Modeling of Point-Line Associations. Wireframes are ubiquitous in man-made environments. The advent of deep neural networks has enabled reliable 2D wireframe prediction from single images [77, 85]. Wireframe representations have also proven beneficial for feature matching, either through junction-aware heuristics [42, 76] or learned approaches [48, 64]. Recent line-based reconstruction systems [39, 40] further incorporate point-line associations in optimization via 3D point-to-line distance regularization. However, these formulations introduce dense couplings between line segments that share junctions, disrupting the sparse structure and leading to expensive Schur complement operations, particularly with dense solvers. Our method addresses this limitation through a reformulation that preserves sparsity while fully exploiting wireframe constraints.

3 A Unified Framework of Holistic Bundle Adjustment

3.1 Review of Classical Bundle Adjustment

Bundle adjustment (BA) jointly optimizes camera parameters and 3D structure by minimizing reprojection errors across all observations [63]. Given N images, each with pose $[R_i \mid \mathbf{t}_i]$ and associated (potentially shared) camera intrinsics $K_{c(i)}$, we denote the full camera parameters of image i as $C_i = (K_{c(i)}, R_i, \mathbf{t}_i)$ and collect them in $\mathcal{C} = \{C_1, \dots, C_N\}$. Given M 3D points $\mathcal{X} = \{\mathbf{X}_1, \dots, \mathbf{X}_M\}$, the classical BA objective is:

$$E(\mathcal{C}, \mathcal{X}) = \sum_{(i,j) \in \mathcal{O}} \|\pi(C_i, \mathbf{X}_j) - \mathbf{x}_{ij}\|^2, \quad (1)$$

where $\pi(C_i, \mathbf{X}_j)$ projects point $\mathbf{X}_j$ into camera C_i , $\mathbf{x}_{ij}$ is the corresponding 2D observation, and $\mathcal{O}$ is the set of all visibility pairs. Minimizing Eq. (1) via the Levenberg-Marquardt algorithm leads to normal equations with a characteristic block structure:

$$\begin{bmatrix} \mathbf{U} & \mathbf{W} \\ \mathbf{W}^\top & \mathbf{V} \end{bmatrix} \begin{bmatrix} \delta\mathcal{C} \\ \delta\mathcal{X} \end{bmatrix} = - \begin{bmatrix} \mathbf{g}_c \\ \mathbf{g}_x \end{bmatrix}, \quad (2)$$

where $\mathbf{U}$ and $\mathbf{V}$ are the camera-camera and point-point blocks of the approximate Hessian, $\mathbf{W}$ is the camera-point off-diagonal, and $\mathbf{g}_c$, $\mathbf{g}_x$ are the gradient vectors. A crucial property is that $\mathbf{V}$ is *block-diagonal*: each 3D point appears only in residuals involving its observing cameras, creating no direct coupling between different points. This enables efficient Schur complement elimination, where points are marginalized to yield a reduced camera-only system:

$$(\mathbf{U} - \mathbf{W}\mathbf{V}^{-1}\mathbf{W}^\top) \delta\mathcal{C} = -\mathbf{g}_c + \mathbf{W}\mathbf{V}^{-1}\mathbf{g}_x. \quad (3)$$

Since $\mathbf{V}^{-1}$ reduces to inverting small independent blocks, this elimination is both efficient and numerically stable. Modern solvers such as Ceres [1] exploit this structure extensively. However, real-world scenes often exhibit richer geometric relations such as parallelism, coplanarity, and wireframe connectivity. We next introduce a taxonomy that distinguishes features from groups and show how both can be incorporated while preserving efficient Schur elimination. Figure 2 presents an illustration of the main concepts introduced in this paper.

3.2 Features and Groups: A Taxonomy on Sparse 3D Structures

We propose a taxonomy that categorizes geometric entities into *features* and *groups* based on their role in the normal equations of bundle adjustment.

Features are geometric primitives with direct 2D measurements in each observing image. Points and lines are the canonical examples: a 3D point $\mathbf{X}_j$ is observed as a 2D keypoint $\mathbf{x}_{ij}$; a 3D line $\mathbf{L}_k$, parameterized in Plücker coordinates [7], is observed as a 2D line segment ℓ_{ik} . Adding lines to Eq. (1) yields:

$$E = \sum_{(i,j) \in \mathcal{O}_p} \|\pi_p(C_i, \mathbf{X}_j) - \mathbf{x}_{ij}\|^2 + \sum_{(i,k) \in \mathcal{O}_l} \|\rho(\pi_l(C_i, \mathbf{L}_k), \ell_{ik})\|^2, \quad (4)$$

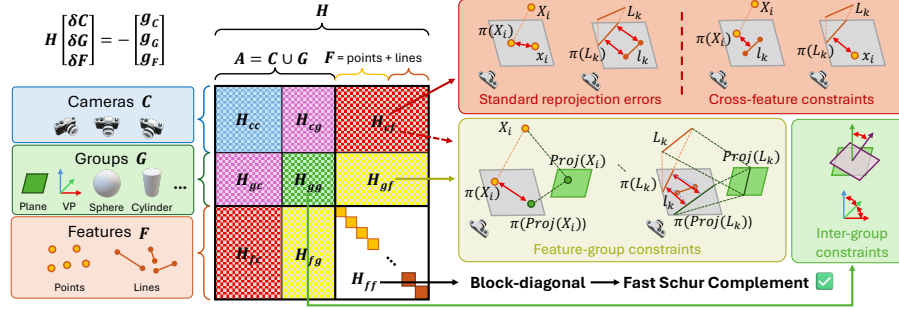


Fig. 2: Taxonomy of features, groups, and cross-feature associations. Left: Features $\mathcal{F}$ (points, lines) have direct 2D measurements; groups $\mathcal{G}$ (planes, VPs, 3D primitives) encode higher-order relations among features. **Middle:** approximate Hessian with block diagonal features $\mathbf{H}_{ff}$ matrix. **Right:** representations of the features, feature-group, cross-feature, and inter-group constraints in our optimization problem.

where $\rho(\cdot, \cdot) \in \mathbb{R}^4$ computes the perpendicular distance vectors between the endpoints of the observed 2D segment and the projected 3D line. In the vanilla point-line bundle adjustment with reprojection errors, each feature contributes an independent diagonal block to $\mathbf{V}$ in the normal equation, since its residuals involve only itself and its observing cameras, without coupling the other features.

Groups encode higher-order geometric relations among features. Vanishing points (VPs) encode parallelism among 3D lines, planes encode coplanarity of points and lines, and other primitives such as spheres, cylinders, cuboids, and object priors constrain features to lie on a parametric manifold. Unlike features, groups generally lack direct per-image 2D measurements in the classical sense. Instead, groups enforce consistency on its associated features: a VP group enforces directional consistency among its associated lines, while a plane group enforces coplanarity among its associated features.

There are two common approaches to incorporating such group constraints into bundle adjustment. The first is explicit reparameterization [6], where features are constrained to lie exactly on a structure, *e.g.*, reducing a 3D point on a plane from 3 to 2 degrees of freedom. However, this requires perfect feature-group associations prior to optimization, which is challenging in practice. The second introduces soft regularization terms [35, 58] that penalize deviations from structural constraints. This can be done either through direct pairwise or higher-order constraints between features (*e.g.*, multiple points should be coplanar), or by introducing explicit group parameters with soft feature-group regularizations.

Computational Structure. The two soft regularization approaches have different implications for the structure of the normal equations. Direct constraints between features introduce off-diagonal blocks in the feature Hessian, breaking the block-diagonal property that enables efficient Schur elimination. For example, enforcing parallelism between two lines without a VP parameter creates a residual depending on both line parameters, filling in off-diagonal blocks and thus coupling them in the Hessian.

In contrast, introducing explicit group parameters avoids this coupling (Fig. 2, middle). Each feature-group residual involves exactly one feature and one group, analogous to how a camera-feature residual involves one feature and one camera. The key is to place group parameters alongside cameras in the non-eliminated block. Given group parameters $\mathcal{G} = \{G_1, \dots, G_P\}$, we form an augmented set $\mathcal{A} = \mathcal{C} \cup \mathcal{G}$ and rewrite Eq. (2) as:

$$\begin{bmatrix} \mathbf{H}_{aa} & \mathbf{H}_{af} \\ \mathbf{H}_{fa} & \mathbf{H}_{ff} \end{bmatrix} \begin{bmatrix} \delta\mathcal{A} \\ \delta\mathcal{F} \end{bmatrix} = - \begin{bmatrix} \mathbf{g}_a \\ \mathbf{g}_f \end{bmatrix}. \quad (5)$$

Since each residual involves at most one feature, $\mathbf{H}_{ff}$ remains block-diagonal. The Schur complement eliminates features to yield a reduced system over cameras and groups: $\mathbf{S} = \mathbf{H}_{aa} - \mathbf{H}_{af}\mathbf{H}_{ff}^{-1}\mathbf{H}_{fa}$.

Note that the categorization of entities into features and groups reflects computational considerations, not just geometric ones. A line could in principle be treated as a group enforcing collinearity among its associated points, but lines typically number in the thousands, and placing them in the non-eliminated block would make the reduced system $\mathbf{S}$ prohibitively large. Similarly, a VP could act as a feature if 2D direction measurements are available, and a plane normal could serve as a feature when estimated from monocular cues. However, such 2D measurements tend to be inaccurate, and groups like planes and VPs are typically few enough (*i.e.*, $|\mathcal{G}| \ll |\mathcal{F}|$) to remain in the non-eliminated block without significant overhead. Nonetheless, when available, 2D measurements of groups are still compatible with our framework, as they contribute only to $\mathbf{H}_{aa}$ and do not affect the block-diagonal structure of $\mathbf{H}_{ff}$.

Inter-Group Relations. A further benefit of placing groups in $\mathcal{A}$ is that cross-group and camera-group constraints can be added at no cost to the Schur complement structure, since they involve only parameters within $\mathcal{A}$ and do not affect $\mathbf{H}_{ff}$. For instance, we employ orthogonality and parallelism constraints between VP pairs and between plane normal pairs to enforce Manhattan world priors, which contribute only to $\mathbf{H}_{aa}$, or more specifically $\mathbf{H}_{gg}$ in Fig. 2.

Cross-Feature Associations (Wireframes). Beyond group-feature relations, features of different types can be directly associated with each other. In architectural and urban scenes, points frequently lie on lines at junctions, forming wireframe structures. These point-line associations encode geometric constraints that are complementary to group constraints: while groups relate features to a shared structural entity, cross-feature associations relate features directly to each other. Unlike group constraints, this coupling between features cannot be resolved through parameter ordering alone, as both entities belong to $\mathcal{F}$. We address this challenge in Sec. 3.4.

3.3 Group Constraints in Bundle Adjustment

Having established that groups can be placed alongside cameras to preserve the block-diagonal structure (Sec. 3.2), we now design the group-feature residuals. For directional groups such as VPs, angular residuals between the line direction

$\mathbf{d}_k$ (from the Plücker representation) and the VP direction $\mathbf{v}_g$ are scale-free and straightforward to integrate:

$$r_v = |\mathbf{d}_k \times \mathbf{v}_g|. \quad (6)$$

Positional groups such as planes are less straightforward. A natural formulation uses direct 3D residuals; for example, coplanarity can be enforced via the point-to-plane distance $r_{3D} = \mathbf{n}^\top \mathbf{X}_j - d$, where $\mathbf{n}$ and d denote the plane normal and offset. However, such 3D constraints are problematic. Because reprojection errors are invariant to global scale, the optimizer can reduce 3D penalties by shrinking the reconstruction. In SfM, scale is typically fixed by constraining the translation of the initial image pair, which does not prevent the remaining structure from collapsing towards it. Moreover, balancing 3D and 2D terms depends on scene scale: proper weighting would require the 3D covariance of the features, which depends on viewing configurations and cannot be recovered from 2D observations alone. In contrast, 2D pixel noise is well characterized and approximately isotropic, making reprojection-based formulations easier to calibrate.

Group-Induced Reprojection Error. Motivated by the discussions above, we propose to formulate positional group constraints through pixel-space residuals that measure the reprojection difference caused by the group projection (Fig. 2, right). For a plane group G_g with normal $\mathbf{n}$ and offset d , we let the group-induced residual for an associated point $\mathbf{X}_j$ observed in camera C_i be:

$$r_g^{(p)} = \pi_p(C_i, \mathbf{X}_j) - \pi_p(C_i, \text{Proj}_{G_g}(\mathbf{X}_j)) \in \mathbb{R}^2, \quad (7)$$

where $\text{Proj}_{G_g}(\mathbf{X}_j) = \mathbf{X}_j - \frac{\mathbf{n}^\top \mathbf{X}_j - d}{\|\mathbf{n}\|^2} \mathbf{n}$ projects the 3D point onto the plane surface. If the point lies on the plane, $\text{Proj}_{G_g}(\mathbf{X}_j) = \mathbf{X}_j$ and the residual vanishes. Otherwise, it measures the pixel displacement between the original and group-constrained reprojections. The same principle applies to associated lines: the residual is the difference between the standard line reprojection residual and that of the plane-projected line:

$$r_g^{(l)} = \rho(\pi_l(C_i, \mathbf{L}_k), \ell_{ik}) - \rho(\pi_l(C_i, \text{Proj}_{G_g}(\mathbf{L}_k)), \ell_{ik}) \in \mathbb{R}^4. \quad (8)$$

More generally, any geometric entity that defines a projection operator from 3D space onto a constraint surface can serve as a group in our framework, with the group-induced reprojection error following the same pattern. In all cases, each residual involves exactly one feature (in $\mathcal{F}$) and at most one group plus one camera (in $\mathcal{A}$), preserving the block-diagonal structure of $\mathbf{H}_{ff}$.

Statistical Interpretation: Group Anchoring through Implicit Weighting. Beyond avoiding scale ambiguity, the reprojection difference formulation provides a clean mechanism for anchoring the group surface through well-observed features. Concretely, let $\boldsymbol{\delta} = \mathbf{X}_j - \text{Proj}_{G_g}(\mathbf{X}_j) \in \mathbb{R}^3$ denote the displacement of a point from the group surface, and let $J_i = \partial \pi_p / \partial \mathbf{X} \big|_{C_i, \mathbf{X}_j} \in \mathbb{R}^{2 \times 3}$ be the projection Jacobian in view i . To first order, the group-induced residual (Eq. (7)) linearizes as:

$$r_g^{(p)} \approx J_i \boldsymbol{\delta}. \quad (9)$$

The effective cost of the group constraint, summed over all N_j views observing the point, is therefore:

$$E_g \approx \sum_i \|J_i \delta\|^2 = \delta^\top \left(\sum_i J_i^\top J_i \right) \delta. \quad (10)$$

The matrix $\sum_i J_i^\top J_i$ is the Fisher information matrix of the point’s 3D position, *i.e.*, the inverse of its covariance $\Sigma_{\mathbf{x}}^{-1}$ [22]. Notably, this is the same matrix that arises from the standard reprojection residuals used to triangulate the point, so the group constraint inherits a geometrically consistent uncertainty model. The group-induced cost is thus equivalent to a Mahalanobis distance $\delta^\top \Sigma_{\mathbf{x}}^{-1} \delta$. We want to note that under the common isotropic noise assumption, our group residual is equivalent to 3D point-to-plane distance with 3D point covariance computed from relinearized reprojection Jacobians at each iteration, which is closely related to the reduction perspective of implicit models in [63].

This directly determines how the group surface is estimated: well-observed features (with large Fisher information) produce large residuals when off-surface, anchoring the group to their positions, while poorly observed features contribute little regardless of displacement. The group parameters are therefore driven by reliable features, which in turn benefit from the structural consistency enforced by the group constraint. As shown in our experiments, this consistently improves both pose and geometry accuracy.

Note that our formulation cannot recover features with degenerate viewing geometry onto the group surface. A 3D formulation could push such features onto the surface, but since degenerate directions are precisely those where reprojection is insensitive, cameras do not benefit from this recovery. Importantly, such features also pose minimal risk to group estimation: their weak Fisher information produces small residuals that have little influence on the group parameters. **Sparsity Analysis on $\mathbf{H}_{cg}$.** While the group-induced reprojection residual involves camera, feature, and group parameters simultaneously, it does not worsen the sparsity pattern of the reduced system compared to a 3D formulation. A 3D point-to-plane residual $r = \mathbf{n}^\top \mathbf{X} - d$ has no direct dependence on camera parameters, contributing only to $\mathbf{H}_{gf}$ and $\mathbf{H}_{ff}$. However, after Schur complement elimination of features, indirect camera-group coupling arises in $\mathbf{H}_{cg}$ wherever a camera observes a feature associated with a group. Our 2D formulation creates this coupling directly, but the sparsity pattern is identical – a nonzero entry in $\mathbf{H}_{cg}$ exists if and only if the camera sees some feature associated with the group. The structure of the reduced system $\mathbf{S}$ is thus unchanged.

3.4 Cross-Feature Constraints in Bundle Adjustment

As discussed in Sec. 3.2, wireframe constraints couple features directly in $\mathbf{H}_{ff}$, breaking its block-diagonal structure. For example, enforcing a point-on-line constraint in 3D as in [40] introduces a residual that depends on both feature parameters, and the 3D point-to-line distance suffers from the same scale-dependent

weight calibration issues as discussed in Sec. 3.3. We therefore propose a reformulation that preserves sparsity.

Cross-Feature Reprojection Error. Our key observation is that wireframe constraints can be reformulated as reprojection errors where only *one* feature participates as an optimization variable, while the *2D observation* of the other feature serves as a fixed measurement. Given a wireframe edge $(\mathbf{X}_j, \mathbf{L}_k)$, we define two complementary residuals across their respective visibility sets.

For an image i where line $\mathbf{L}_k$ is observed as ℓ_{ik} but point $\mathbf{X}_j$ is *not* directly detected, the point-to-line residual measures the distance from the projected point to the observed line:

$$r_{p \rightarrow l} = d(\pi_p(C_i, \mathbf{X}_j), \ell_{ik}), \quad (11)$$

where $d(\cdot, \cdot)$ is the perpendicular point-to-line distance in pixels. The optimization variables are C_i (camera, in $\mathcal{A}$) and $\mathbf{X}_j$ (feature, in $\mathcal{F}$); the 2D line observation ℓ_{ik} enters as a constant measurement. Intuitively, if the point lies on the line, it should project near the observed line in any view where the line is visible, even in views where the point itself is not detected. As with the group-induced reprojection error (Sec. 3.3), this 2D formulation avoids scale-dependent 3D distances and implicitly weights the constraint by observation geometry, with well-observed features providing stronger anchoring signals (Eq. (10)).

Conversely, for an image i where point $\mathbf{X}_j$ is observed as $\mathbf{x}_{ij}$ but line $\mathbf{L}_k$ is not detected, the line-to-point residual is:

$$r_{l \rightarrow p} = d(\mathbf{x}_{ij}, \pi_l(C_i, \mathbf{L}_k)), \quad (12)$$

where $\mathbf{x}_{ij}$ is a constant measurement and $\mathbf{L}_k$ is the optimization variable. In images where both features are directly observed, we omit the cross-feature terms as their standard reprojection residuals (Eq. (4)) already provide the constraint. Note that this cross-feature reprojection error formulation inherits the same statistical interpretation of the classical reprojection error and thus naturally supports more advanced noise models.

Sparsity Analysis. Each cross-feature residual involves exactly one feature and one camera. The Jacobian of $r_{p \rightarrow l}$ has nonzero blocks only for C_i and $\mathbf{X}_j$; that of $r_{l \rightarrow p}$ only for C_i and $\mathbf{L}_k$. No off-diagonal blocks arise between point j and line k in $\mathbf{H}_{ff}$, preserving its block-diagonal structure. The constraint is mediated through the 2D observation of the partner feature, decoupling the features while still enforcing their geometric relation. Note that cross-feature constraints extend the effective visibility graph, making the reduced system $\mathbf{S}$ denser in camera-camera blocks, but efficient Schur elimination remains preserved.

3.5 Discussions and Applications

Convergence and Termination. A practical benefit of formulating constraints in pixel space is simplified convergence monitoring. When mixing 3D regularization terms (in metric units) with 2D reprojection errors (in pixels), the residuals have different magnitudes and convergence dynamics, complicating cost-based

termination criteria such as the relative cost change $|\Delta E|/E$. With our unified pixel-space formulation, the total cost has a consistent interpretation and such criteria apply directly. For the remaining angular residuals (*e.g.*, VP constraints in Eq. (6) and cross-group angular constraints), we isolate their contribution and monitor convergence based on the reprojection terms only.

Structure-from-Motion Pipeline. Our bundle adjustment framework integrates naturally into incremental structure-from-motion. After registering each new image and triangulating features and groups, the system performs local bundle adjustment with the full set of constraints, including features, groups, and wireframe edges. Image selection leverages both point and line visibility, and camera registration uses both point and line correspondences. This yields an end-to-end SfM pipeline where geometric structures participate from the earliest stages of reconstruction. We also want to note that, the sparsity analysis and Schur elimination strategy extend directly to global positioning [47, 86] in global SfM, as cameras, features, and groups share the same computational structure.

Discussion: Robustness to Noisy Upstream Associations. Our work focuses on optimization once structural priors are available, rather than solving the problem of adaptive model selection. We therefore employ an structure association strategy that favors precision over recall and optimize these residuals with robust losses. Importantly, robust thresholding becomes easier in our pixel-level formulation than in mixed 2D/3D formulations, especially for SfM with its scale ambiguity. Nonetheless, the reliance on reasonably good quality of associations is an important practical limitation of our framework. We also want to note the distinction between measurement noise and association reliability, where the former is handled in a principled way through how our residuals are formulated, while the latter is closely related to model selection and benefits from future advances in estimating confidences of the extracted structural constraints.

4 Experiments

Implementation Details. Our framework is agnostic to the choice of feature and group detection and matching methods. We use ALIKED [82] / DeepLSD [49] and LightGlue [38] / GlueStick [48] for point / line detection and matching. Vanishing points are detected with JLinkage [61], and planes with a custom segmentation method fitting planes to predicted depth and surface normals of MoGe-2 [68, 69]. Images are resized so that the longer side is at most 800 pixels, to satisfy model constraints. Within each image, feature-group associations are obtained during detection (*e.g.*, lines assigned to VPs, points and lines to plane segments), while point-line associations are derived from spatial incidences. Across images, group matchings are established either by voting over matched features or by IoU checks of warped segmentation masks via dense matching [18, 81]. To construct 3D constraints from 2D observations, we aggregate feature-group and cross-feature associations across images through voting following [40]. Group constraints count images where a feature observation is associated with a group, while wireframe constraints count images where a point-line pair forms a junc-

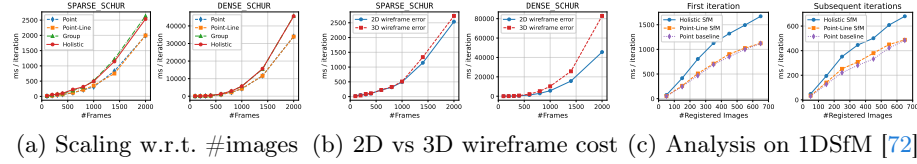


Fig. 3: Bundle adjustment runtime analysis. (a) Per-iteration cost scaling with number of images. (b) Comparison between 2D vs 3D wireframe cost. (c) First iteration vs. subsequent iterations of BA on 1DSfM, averaged across 3 scenes.

Table 1: Runtime and reconstruction statistics on 1DSfM [72]. The *Point baseline* is equivalent to COLMAP with only one change: replacing the gradient-norm termination criterion with the relative cost change based criterion. The two other methods use the same criterion as the point baseline. We report runtime, BA statistics, reconstruction size (#Reg: number of registrations, #pts: 3D points, #lns: lines, #grps: group primitives), and mean reprojection error over 3D point observations (in pixels).

Scene	Method	Runtime & Convergence				Reconstruction statistics				
		SfM (min)	ms/iter	iter/BA	#BA	#Reg	#pts/#obs	#lns	#grps	err (px)
Tower of London	COLMAP reference	63.6	—	16.2	194	935 / 1322	97k/731k	—	—	1.08
	Point baseline	46.4	132.5	4.8	196	909 / 1322	99k/731k	—	—	1.04
	Point-Line SfM	54.6	158.9	4.8	205	927 / 1322	99k/732k	2.9k	—	1.02
	Holistic SfM (Ours)	59.2	214.0	11.8	226	963 / 1322	99k/731k	2.8k	117	1.03
Madrid Metropolis	COLMAP reference	45.3	—	21.7	225	742 / 1061	59k/462k	—	—	1.13
	Point baseline	31.5	136.3	5.4	233	752 / 1061	65k/476k	—	—	1.07
	Point-Line SfM	34.8	153.5	5.6	249	753 / 1061	63k/465k	2.1k	—	1.05
	Holistic SfM (Ours)	37.6	192.1	9.5	253	750 / 1061	65k/469k	2.1k	42	1.06
Gendarmenmarkt	COLMAP reference	68.8	—	14.6	234	943 / 1173	85k/846k	—	—	1.16
	Point baseline	43.5	281.5	4.2	248	950 / 1173	89k/847k	—	—	1.10
	Point-Line SfM	45.7	274.7	4.5	266	947 / 1173	90k/851k	3.8k	—	1.10
	Holistic SfM (Ours)	63.5	394.3	11.3	281	951 / 1173	88k/846k	3.7k	91	1.10

tion. Constraints are instantiated when the number of such images exceeds 3, and subsequently optimized with a robust Cauchy loss. Please refer to more details in our supplementary material.

Our proposed BA is built on top of Ceres solver [1] with analytical Jacobians. The SfM pipeline follows COLMAP [53], which automatically selects the numerical solver depending on number of images. Runtime experiments are conducted on a workstation with an 8-core Intel Core i7-10700K CPU at 3.80 GHz.

4.1 Benchmarking Holistic Bundle Adjustment

Synthetic Benchmark. To analyze scaling behavior in isolation, we construct synthetic BA problems by varying the number of images from 100 to 2000. For each configuration, we generate points, lines, planes, and wireframe edges with counts scaling proportionally. Each feature is observed with an average track length of 4-8 images. Please refer to details in supplementary materials.

We compare four BA configurations of increasing complexity: *Point* (point-only), *Point-Line* (points and lines), *Groups* (adding group constraints from Sec. 3.3), and *Holistic* (further adding wireframe constraints from Sec. 3.4). Figure 3(a) shows per-iteration cost as a function of problem sizes. All bundle adjustment configurations scale similarly under both *SPARSE_SCHUR* ($\sim n^{1.6-1.9}$) and *DENSE_SCHUR* ($\sim n^{2.6-2.8}$), indicating that our formulation preserves the

Table 2: Geometry evaluation on Hypersim [50] (8 scenes). Points and lines are measured against ground-truth meshes.

	Points				Lines					
	Inlier (%) $\uparrow$			Med. $\downarrow$ (mm)	Length Recall (m) $\uparrow$			Precision (%) $\uparrow$		
	@1mm	@5mm	@10mm		@1mm	@5mm	@10mm	@1mm	@5mm	@10mm
Per-point/per-line refinement	9.3	31.7	45.6	15.10	72.1	222.5	301.2	21.3	59.4	77.3
+ Group constraints	11.5	35.6	48.5	13.80	93.2	248.8	315.5	27.3	65.5	81.0
+ Group and wireframe constraints	11.9	36.3	48.9	13.55	93.3	249.5	315.7	27.3	65.5	81.0

Table 3: Point reconstruction quality on ETH3D [54] (13 scenes). Reported against laser scans using the official ETH3D multi-view evaluation benchmark.

	Accuracy (%) $\uparrow$				Completeness (%) $\uparrow$				F1 (%) $\uparrow$			
	@1cm	@2cm	@5cm	@10cm	@1cm	@2cm	@5cm	@10cm	@1cm	@2cm	@5cm	@10cm
Per-point refinement	45.17	58.70	75.53	84.57	0.14	0.74	4.10	11.09	0.29	1.46	7.66	19.25
+ Group and wireframe constraints	47.05	60.59	76.84	85.40	0.15	0.76	4.18	11.21	0.30	1.50	7.81	19.47

asymptotic complexity of classical BA. With DENSE_SCHUR, runtime is strictly determined by the size of the reduced system, with *Point* and *Point-Line*, *Groups* and *Holistic* having nearly identical cost. This supports our analysis that groups behave similarly to camera variables in the normal equations. Figure 3(b) compares the per-iteration runtime of our 2D reprojection-based wireframe formulation with a 3D endpoint-distance formulation. With SPARSE_SCHUR, the 2D formulation is moderately faster as it avoids direct coupling between features. With DENSE_SCHUR, the gap widens because the 3D formulation produces significantly denser fill-in in the reduced system.

Real-World Benchmark. We further evaluate three large-scale scenes from 1DSfM [72] within a full incremental SfM pipeline (Tab. 1). As discussed in Sec. 3.5, we use the relative cost change as the BA termination criterion, which empirically stops earlier than COLMAP, whose termination relies on the gradient norm. Our method registers a comparable or slightly larger number of images while maintaining similar reprojection errors. The higher iteration count per BA call (e.g., 11.8 vs. 4.8 on Tower of London) reflects the richer constraint set where structural constraints generally require more iterations for convergence, but the overall pipeline runtime remains $\sim 1.3\times$ of the point baseline, substantially lower than the $2\text{--}4\times$ overhead over COLMAP reported in [39]. Figure 3(c) shows the BA cost breakdown on 1DSfM. The first iteration of each BA call performs symbolic analysis (sparsity pattern computation and ordering), while subsequent iterations require only numeric factorization. This symbolic overhead is amortized across iterations, after which the steady-state numeric cost dominates.

4.2 Applications

Geometry Evaluation. To study the benefits of group and wireframe constraints on the resulting 3D maps, we evaluate reconstruction quality against ground-truth geometry on the first eight scenes of Hypersim [50], following the protocol of [39]. Points and lines are measured against ground-truth meshes. For

Table 4: Evaluating the effectiveness of bundle adjustment in full structure-from-motion pipelines. Camera pose accuracy is measured by relative pose AUCs.

	Hypersim (8 scenes)			ScanNet++ (20 scenes)		
	AUC@3°	AUC@5°	AUC@10°	AUC@3°	AUC@5°	AUC@10°
COLMAP	89.3	90.4	92.2	84.0	86.0	87.5
Point-Line SfM	92.2	93.2	94.0	84.7	86.7	89.2
Holistic SfM (Ours)	92.5	93.4	94.2	87.4	89.4	90.6

	ETH3D (11 scenes)			7Scenes (7 scenes)		
	AUC@3°	AUC@5°	AUC@10°	AUC@3°	AUC@5°	AUC@10°
COLMAP	50.2	66.4	77.6	22.9	48.7	75.5
Point-Line SfM	49.4	67.1	79.6	23.1	48.9	76.8
Holistic SfM (Ours)	50.4	68.1	80.4	24.3	50.8	78.4

Table 5: Ablation of the group-induced 2D reprojection error vs direct 3D constraints via geometry evaluation on ETH3D and SfM evaluation on ScanNet++.

Method	ETH3D Acc. (%) $\uparrow$			ScanNet++ AUC $\uparrow$			
	@1cm	@5cm	@10cm	@1°	@3°	@5°	@10°
Point BA	45.17	75.53	84.57	66.1	84.0	86.0	87.5
Holistic BA w/. 3D group error ($w=1$)	45.17	75.55	84.58	67.4	85.0	87.1	89.1
Holistic BA w/. 3D group error ($w=10$)	45.24	75.61	84.63	68.1	85.0	89.0	90.5
Holistic BA w/. 3D group error ($w=100$)	45.72	76.12	84.88	<u>68.3</u>	87.8	89.9	91.1
Holistic BA w/. 3D group error ($w=1000$)	46.69	76.71	<u>85.15</u>	67.1	85.1	87.1	88.3
Holistic BA (Ours)	47.05	76.84	85.40	69.4	<u>87.4</u>	<u>89.4</u>	<u>90.6</u>

points, we report the inlier ratio (percentage within threshold) and median distance. For lines, length recall is the average correct length in meters across scenes, while precision is the fraction of total reconstructed length within threshold. The results, reported in Tab. 2, show that group constraints consistently improve the accuracy of reconstructed 3D points and substantially increase both the recall and precision of line segments. Incorporating additional wireframe constraints further improves performance by anchoring points to structural lines, thereby introducing stronger geometric consistency. We also evaluate our structural constraints on ETH3D [54] using the official multi-view benchmark in Tab. 3. Our method improves both accuracy and completeness across all thresholds, demonstrating the effectiveness of group constraints in enhancing feature reconstruction by enforcing structural consistency.

Structure from Motion. As discussed in Sec. 3.5, we integrate our bundle adjustment (BA) framework into a full incremental SfM pipeline and evaluate camera pose accuracy on four datasets: Hypersim [50], ScanNet++ [79], ETH3D [54], and 7Scenes [55]. We use the first eight scenes of Hypersim as in [39] and adopt the splits from [37] for the remaining datasets. Our holistic bundle adjustment consistently improves pose accuracy over prior methods across all datasets, covering both indoor and outdoor scenes (Tab. 4). The improvement is particularly pronounced on ScanNet++, which contains indoor environments with rich geometric structures such as parallel and coplanar surfaces.

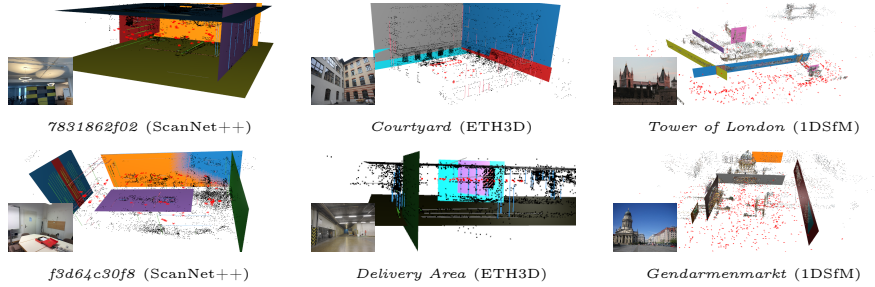


Fig. 4: Visualizations of bundle adjusted holistic 3D structures. Parallel lines are colored the same. Note that lines and planes are optimized as infinite entities in the bundle adjustment, and their extents in the figures are only for illustration purpose.

Ablation study. We report results in Tab. 5, comparing our 2D group-induced reprojection error with direct 3D plane-distance constraints under different weights. The 3D formulation requires careful tuning: small weights underuse structural information, while large weights ($w=1000$) degrade pose accuracy on ScanNet++. In contrast, our 2D formulation requires no tuning and achieves the best or competitive results on both geometry (ETH3D) and pose (ScanNet++) metrics.

Mapping Geometric Primitives. Beyond quantitative improvements, our framework produces semantically richer maps with explicit geometric primitives. Example holistic 3D reconstructions are shown in Fig. 4. While planes and vanishing points are the most common structural groups in real-world scenes, our formulation is general and supports other primitives. In Fig. 1, we show bundle adjustment on a scene containing planes, spheres, and cylinders, where groups are detected via SAM3 [8] with semantic prompts (football→sphere, can→cylinder, laptop/postcard→plane). This demonstrates its flexibility and shows the potential of combining geometric abstraction with modern visual-language models. We believe our framework opens up opportunities to further integrate geometric reasoning with visual-language understanding.

5 Conclusion

We present a unified bundle adjustment framework that integrates higher-order geometric structures while preserving efficiency. By placing group parameters alongside cameras and formulating all constraints as reprojection-based residuals, we maintain efficient Schur elimination and avoid the scale ambiguity and conditioning issues of 3D regularization. Experiments demonstrate comparable runtime with significantly richer reconstructions and improved accuracy.

Acknowledgements. Viktor Larsson was supported by ELLIIT and the Swedish Research Council (Grant No. 2023-05424). We sincerely thank Alexander Veicht, Paul-Edouard Sarlin, Eric Dexheimer, Xinyue Zhang, Linfei Pan, Elisabetta Fedele, Johannes Schönberger, Lixin Xue, and anonymous reviewers for their helpful feedbacks.

Appendix

This supplementary material accompanies the main paper and is organized as follows. We begin with implementation details for image description, matching, and structure-from-motion, as well as the synthetic benchmark setup in Appendix A. We then present some example groups and their parameterizations in Appendix B, followed by more details and discussions on the proposed bundle adjustment framework in Appendix C. Finally, we show some additional visualizations.

A Implementation Details

A.1 Additional Details on Image Description and Matching

Feature Detection and Matching. We use ALIKED [82] for point detection and LightGlue [38] for point matching. For lines, we use DeepLSD [49] for detection and GlueStick [48] for matching. Vanishing points are detected with JLinkage [61], and planes are segmented by fitting to predicted depth and surface normals from MoGe-2 [68,69]. Wireframe junctions are derived from spatial incidences between detected points and lines within a distance threshold of 2 pixels. Figure 5 shows example detection results, where the junctions are points associated with more than one line, connecting them into a wireframe structure. Images are resized so that the longer side is at most 800 pixels.

Cross-Image Group Matching. Groups detected in different images must be matched to establish 3D constraints. We support two options. The first is voting via matched features: for each pair of groups in two images, we count the number of matched features associated with both groups and establish a match if this count exceeds 3. The second is matching via dense correspondence: we warp the segmentation mask from one image to another using RoMa [18], and compute the overlap ratio between the warped mask and candidate group masks in the target image, matching groups with overlap ratio above 0.6 and intersection area above 1000 pixels. Note that non-segmentation-based groups such as VPs can only use voting. Empirically, voting yields higher matching accuracy due to its reliance on verified feature correspondences, while dense matching based warping offers the advantage of matching points, lines, and segmentation-based groups jointly.

Geometric Verification. When prior camera poses are available, we verify group associations through geometric consistency checks. For VPs, we back-project 2D image directions into 3D and check angular consistency across views, rejecting matches with angular deviation above 10 degrees. For planes, we compare monocular surface normal estimates with the triangulated plane normal, rejecting matches with angular deviation above 30 degrees.

A.2 Additional Details on Structure-from-Motion

Image Retrieval. For image pair selection, we use NetVLAD [3] to retrieve the top 30 nearest neighbors for each image.

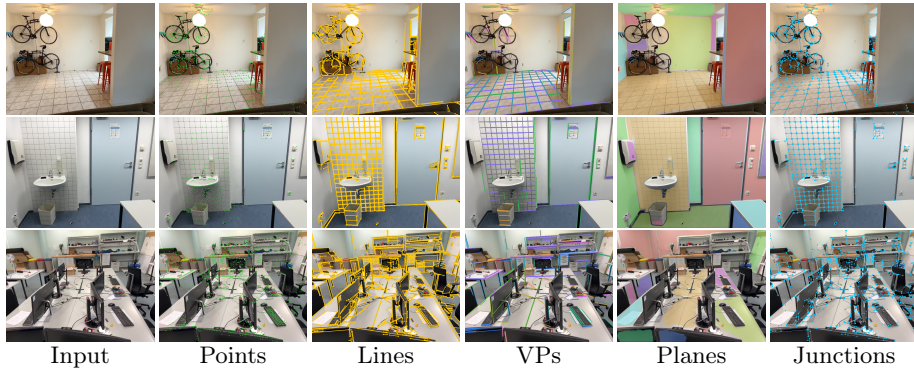


Fig. 5: Example features, groups, and wireframes from image description.

Incremental Mapping. We follow prior work on incremental triangulation of points [53] and lines [39, 40] and extend it to incremental group triangulation as new images are registered. For each newly registered image, point and line features are first triangulated following the standard incremental pipeline. Group triangulation is then performed based on the resulting triangulated features. When a new 2D group observation is introduced, we count the number of shared triangulated features with existing 3D groups. If at least two features overlap, the observation is associated with the corresponding group. If no match is found, the observation is temporarily stored until a consistent set of features emerges. Specifically, when at least three features consistently co-occur across unmatched observations, a new 3D group is initialized. Group parameters are estimated using LO-RANSAC [13]. We adopt a relaxed 3D threshold at this stage and defer fine-grained group refinement to the subsequent optimization. After new triangulations, we check for potential group merges. Two groups of the same type become merge candidates when their 2D observations are connected through feature correspondences across images. A merge is performed if the group parameters are geometrically compatible—for example, vanishing point (VP) directions must lie within an angular threshold—and if the merged parameters do not significantly increase the fitting error of either group’s features. When a merge occurs, feature associations are combined and the group parameters are re-estimated from the aggregated 3D features. Finally, we apply geometric consistency filtering. Groups are validated using a reprojection consistency check similar to the group-induced reprojection error. In addition, we apply a 3D distance filter based on three times the median absolute deviation (MAD). A group is retained only if it contains at least three inlier features and achieves an inlier ratio greater than 0.5.

Reliability-Aware Refinement. Following prior work [39], we classify feature tracks as reliable or unreliable based on their number of active observations and estimated uncertainty. Refinement proceeds in two stages: a full optimization including cameras, reliable features, groups, and wireframe constraints, followed by a fixed-pose refinement of unreliable features. This prevents weakly-constrained features from corrupting pose estimates while retaining them for future use as

more observations accumulate. We extend this mechanism to groups: group associations are updated after each stage, and groups with insufficient active support are excluded from the first stage.

Full Pipeline. Our incremental SfM pipeline is built on top of COLMAP [53] and closely follows its principles, including next image selection, registration with structure-less fallback, triangulation, local bundle adjustment after each image, and periodic global bundle adjustment. We extend the pipeline by integrating group triangulation and structure-aware bundle adjustment at each refinement stage.

A.3 Details on Setup of the Synthetic Benchmark

We construct synthetic bundle adjustment problems to analyze scaling behavior in isolation. Cameras are placed on a sphere surrounding a cubic scene (half-extent 3.0) with viewing directions pointing inward. Within the cube, we generate Manhattan-aligned planes with normals along the x , y , or z axes at random offsets. Points and lines are distributed both on these planes and as free-floating features. Wireframe junctions are created by sampling a regular grid of points on each plane and connecting them with lines along the grid rows and columns, forming point-line associations that model structural corners and edges. The number of images varies from 100 to 2000, with feature counts scaling proportionally: $10N$ points, N lines, and $N/5$ planes for N images, plus $3N$ wireframe edges. Each feature is observed by 4 to 8 cameras on average, with lines on the same plane sharing approximately 50% of their views to model real co-visibility patterns. We add Gaussian noise to 2D observations (1.0 pixel for points and line endpoints in 2D), initial camera poses (1.0 degree for rotations, 0.05 for translations), and initial 3D estimates (0.1 for points and line endpoints in 3D). For fair runtime comparison, we run each BA configuration for exactly 20 iterations.

B Example Groups and Their Parameterizations

We support various types of geometric groups that constrain 3D features (points and lines) to lie on specific surfaces or follow directional constraints. Table 6 summarizes each group with an illustration, degrees of freedom, and the features it constrains. Positive quantities such as radii, semi-axes, and angles are optimized in log-space to ensure positivity during optimization. Below we describe the parameterization for each group type. Note that while Table 6 lists the group types currently implemented, the framework is not limited to these and can be extended to support additional geometric primitives such as splines, parametric surfaces, object-level priors, and parameterized priors from 3D foundation models (e.g., monocular depths and normals).

Vanishing Point (VP). A VP is a point at infinity representing a 3D direction, parameterized as $\mathbf{v} \in \mathbb{S}^2$. The residual for a line with direction $\mathbf{d}$ measures angular deviation:

$$e = \|\mathbf{d} \times \mathbf{v}\|. \quad (13)$$

Table 6: Supported group types with illustrations. Positive quantities (radii, semi-axes, angles) are parameterized in log-space to ensure positivity.

Sketch	Group	#DoF	Features	Geometric Interpretation
	VP	2	Lines	Direction $\mathbf{v} \in \mathbb{S}^2$
	Plane	3	Points, Lines	Normal $\mathbf{n} \in \mathbb{S}^2$, offset w
	Sphere	4	Points	Center $\mathbf{c} \in \mathbb{R}^3$, radius $r > 0$
	Cylinder	5	Points	Axis ($SO(3) \times SO(2)$), radius $r > 0$
	Cone	6	Points	Apex $\mathbf{a} \in \mathbb{R}^3$, axis $\mathbf{d} \in \mathbb{S}^2$, half-angle α
	Ellipsoid	9	Points	$R \in SO(3)$, center $\mathbf{c} \in \mathbb{R}^3$, semi-axes $s_x, s_y, s_z > 0$
	Cuboid	9	Points, Lines	$R \in SO(3)$, 6 face offsets

This is equivalent to $e = \|\mathbf{v} - \text{Proj}_{\mathbf{d}}(\mathbf{v})\|$ where $\text{Proj}_{\mathbf{d}}(\mathbf{v}) = (\mathbf{v}^\top \mathbf{d})\mathbf{d}$, *i.e.* the norm of the component of $\mathbf{v}$ orthogonal to $\mathbf{d}$. Geometrically, both measure $\sin \theta$ where θ is the angle between the two unit vectors.

Plane. A plane is parameterized as $(\mathbf{n}, w)$ where $\mathbf{n} \in \mathbb{S}^2$ is the normal and $w \in \mathbb{R}$ is the signed distance to the origin. The plane equation is $\mathbf{n}^\top \mathbf{x} + w = 0$. Points are projected via:

$$\mathbf{p}_{\text{proj}} = \mathbf{p} - (\mathbf{n}^\top \mathbf{p} + w)\mathbf{n}. \quad (14)$$

For lines in Plücker coordinates $(\mathbf{d}, \mathbf{m})$, the projected direction removes the normal component:

$$\mathbf{d}' = \mathbf{d} - (\mathbf{d} \cdot \mathbf{n})\mathbf{n}. \quad (15)$$

The projected moment is:

$$\mathbf{m}' = \left(1 - \frac{\alpha^2}{\|\mathbf{d}\|^2}\right) \mathbf{m} + \frac{\alpha\gamma}{\|\mathbf{d}\|^2} \mathbf{d} + \left(w - \frac{\delta}{\|\mathbf{d}\|^2}\right) (\mathbf{d} \times \mathbf{n}), \quad (16)$$

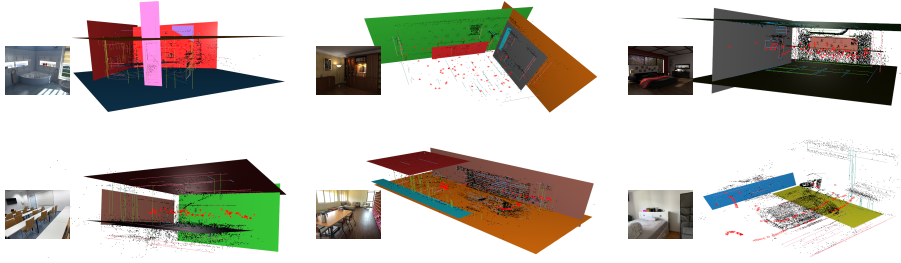
where $\alpha = \mathbf{d} \cdot \mathbf{n}$, $\gamma = \mathbf{m} \cdot \mathbf{n}$, and $\delta = (\mathbf{d} \times \mathbf{n}) \cdot \mathbf{m}$.

Sphere. A sphere is parameterized by its center $\mathbf{c} \in \mathbb{R}^3$ and radius $r > 0$. Points are projected radially:

$$\mathbf{p}_{\text{proj}} = \mathbf{c} + r \cdot \frac{\mathbf{p} - \mathbf{c}}{\|\mathbf{p} - \mathbf{c}\|}. \quad (17)$$

Table 7: Schur complement density analysis. D denotes the fraction of camera pairs with non-zero blocks. Wireframe constraints (WF) only add modest overhead.

Scene	#img	#pts	#lns	#wf	D w/o. WF	D w/. WF	$\Delta\%$
Gendarmenmarkt	951	88k	3.7k	12.0k	41.1%	42.0%	2.2
Madrid Metropolis	750	65k	2.1k	6.0k	23.8%	24.3%	1.9
Tower of London	963	99k	2.8k	9.1k	12.3%	13.2%	7.4
Average					25.7%	26.5%	3.8

**Fig. 6: Additional visualizations of bundle adjusted holistic 3D structures** on Hypersim [50] (top) and ScanNet++ [79] (bottom). Parallel lines are colored the same. Note that lines and planes are optimized as infinite entities in the bundle adjustment, and their extents in the figures are only for illustration purpose.

Cylinder. A cylinder is defined by an infinite axis (parameterized via $SO(3) \times SO(2)$) and radius $r > 0$. The axis is represented using minimal Plücker coordinates [7]. Points are projected radially onto the cylinder surface perpendicular to the axis.

Cone. A cone is parameterized by its apex $\mathbf{a} \in \mathbb{R}^3$, axis direction $\mathbf{d} \in \mathbb{S}^2$, and half-angle $\alpha \in (0, \frac{\pi}{2})$. Points are projected onto the cone surface along the direction perpendicular to the cone’s generatrix.

Ellipsoid. An ellipsoid is parameterized by its pose (rotation $R \in SO(3)$ and center $\mathbf{c}$) and semi-axes (s_x, s_y, s_z) with $s_i > 0$. The surface equation in the local frame is:

$$\mathbf{x}^\top \text{diag}(s_x^{-2}, s_y^{-2}, s_z^{-2}) \mathbf{x} = 1. \quad (18)$$

Points are projected via scaled-radial projection: the point is transformed to the local frame, scaled to a unit sphere, normalized, scaled back, and transformed to the world frame.

Cuboid. A cuboid is parameterized by a rotation $R \in SO(3)$ and six face offsets $d_0, \dots, d_5$ [24]. The rotation defines three orthogonal face normal directions $\pm R_{:,0}, \pm R_{:,1}, \pm R_{:,2}$, and each face i is a plane with normal $\mathbf{n}_i$ and offset d_i . For point projection, we compute the signed distance to each of the six faces and project the point onto the nearest face. For line projection, we first find the closest point on the infinite line to the cuboid center, then determine which face is nearest to this representative point, and finally project the line onto that face using plane projection.

C More Details and Discussions

Balancing Different Types of Costs. As discussed in the main paper, 3D group costs (e.g., point-to-plane distance) and wireframe costs (e.g., 3D point-to-line distance) are problematic because their magnitude depends on the global scene scale, and varies across regions depending on viewing geometry. In contrast, the angular cost for VP constraints, $e = \|\mathbf{d} \times \mathbf{v}\|$, is scale-free: it measures the sine of the angle between unit vectors and does not depend on the global scale of the reconstruction. This makes angular residuals well-suited for directional constraints without the calibration difficulties of 3D metric residuals. However, angular residuals are dimensionless and not directly comparable to pixel-based reprojection errors. We therefore exclude angular terms from the termination criteria and monitor convergence based on reprojection residuals only, as described in the main paper. For optimization, we apply a weight of 5×10^3 to VP angular constraints and use Cauchy loss with scale 0.2. For reprojection-based group and wireframe costs (measured in pixels), we use weights of 0.5 and 0.1 respectively, relative to the base point and line reprojection weights of 1.0.

Relation to Manhattan/Atlanta World Assumption. The Manhattan world assumption [14] posits three mutually orthogonal dominant directions, while the Atlanta world [52] relaxes this to a vertical direction with multiple horizontal directions. Our framework implicitly enforces these assumptions through the combination of feature-group and inter-group constraints. The VP-line constraints align lines with their associated vanishing points, while inter-group orthogonality constraints enforce perpendicularity between VP pairs. Together, these naturally encode the Manhattan or Atlanta structure without requiring explicit world assumption handling.

Effects of Cross-Feature Constraints on the Schur Complement $\mathbf{S}$. As noted in the main paper, cross-feature (wireframe) constraints extend the effective visibility graph, making the Schur complement $\mathbf{S}$ denser in camera-camera blocks. However, for a wireframe edge, new correlations between cameras C_i and C_j only arise when the two cameras share no co-observed features, and the associated point and line each appear in exactly one of them. In practice, this additional density is modest. Table 7 analyzes this on three scenes from the 1DSfM dataset, showing that wireframe constraints increase the Schur complement density by only 2–7% relative to the baseline.

More Qualitative Results. We show some additional visualizations of the bundle adjusted 3D structures in Fig. 6.

References

1. Agarwal, S., Mierle, K.: Ceres solver. <http://ceres-solver.org> 3, 5, 12
2. Agarwal, S., Snavely, N., Seitz, S.M., Szeliski, R.: Bundle adjustment in the large. In: ECCV (2010) 1, 3
3. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: CVPR (2016) 16
4. Bai, X., Cui, H., Shen, S.: Consistent 3d line mapping. In: ECCV (2024) 4

5. Bartoli, A., Coquerelle, M., Sturm, P.: A framework for pencil-of-points structure-from-motion. In: ECCV (2004) [4](#)
6. Bartoli, A., Sturm, P.: Constrained structure and motion from multiple uncalibrated views of a piecewise planar scene. *International Journal of Computer Vision* **52**(1), 45–64 (2003) [4](#), [6](#)
7. Bartoli, A., Sturm, P.: Structure-from-motion using lines: Representation, triangulation, and bundle adjustment. *Computer Vision and Image Understanding (CVIU)* **100**(3), 416–441 (2005) [5](#), [20](#)
8. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., et al.: Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719* (2025) [15](#)
9. Chen, C., Geneva, P., Peng, Y., Lee, W., Huang, G.: Monocular visual-inertial odometry with planar regularities. In: ICRA. pp. 6224–6231 (2023) [4](#)
10. Chen, D., Wang, S., Xie, W., Zhai, S., Wang, N., Bao, H., Zhang, G.: Vip-slam: An efficient tightly-coupled rgb-d visual inertial planar slam. In: ICRA. IEEE (2022) [4](#)
11. Chen, W., Zhang, G., Wimbauer, F., Wang, R., Araslanov, N., Vedaldi, A., Cremers, D.: Back on track: Bundle adjustment for dynamic scene reconstruction. In: ICCV (2025) [3](#)
12. Chen, Y., Davis, T.A., Hager, W.W., Rajamanickam, S.: Algorithm 887: Cholmod, supernodal sparse cholesky factorization and update/downdate. *ACM Transactions on Mathematical Software (TOMS)* **35**(3), 1–14 (2008) [3](#)
13. Chum, O., Matas, J., Kittler, J.: Locally optimized ransac. In: Joint Pattern Recognition Symposium (2003) [17](#)
14. Coughlan, J., Yuille, A.L.: The manhattan world assumption: Regularities in scene statistics which enable bayesian inference. In: NeurIPS (2000) [4](#), [21](#)
15. Davis, T.A.: Algorithm 1000: Suitesparse: Graphblas: Graph algorithms in the language of sparse linear algebra. *ACM Transactions on Mathematical Software (TOMS)* **45**(4), 1–25 (2019) [3](#)
16. Dellaert, F.: Factor graphs and gtsam: A hands-on introduction. Georgia Institute of Technology, Tech. Rep **2**(4) (2012) [3](#)
17. Demmel, N., Sommer, C., Cremers, D., Usenko, V.: Square root bundle adjustment for large-scale reconstruction. In: CVPR (2021) [3](#)
18. Edstedt, J., Sun, Q., Bökman, G., Wadenbäck, M., Felsberg, M.: Roma: Robust dense feature matching. In: CVPR (2024) [11](#), [16](#)
19. Engel, J., Schöps, T., Cremers, D.: Lsd-slam: Large-scale direct monocular slam. In: ECCV (2014) [3](#)
20. Fan, T., Ortiz, J., Hsiao, M., Monge, M., Dong, J., Murphey, T.D., Mukadam, M.: Daba: Decentralized and accelerated large-scale bundle adjustment. *The International Journal of Robotics Research* **44**(10–11), 1892–1919 (2025) [3](#)
21. Faugeras, O.D., Lustman, F.: Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence* **2**(03), 485–508 (1988) [4](#)
22. Förstner, W., Wrobel, B.P.: Photogrammetric computer vision (2016) [9](#)
23. Geneva, P., Eckenhoff, K., Yang, Y., Huang, G.: Lips: Lidar-inertial 3d plane slam. In: IROS. IEEE (2018) [4](#)
24. Hanning, G., Åström, K., Larsson, V.: Pixcuboid: Room layout estimation from multi-view featuremetric alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7224–7233 (2025) [20](#)
25. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003) [3](#)

26. Hartley, R.I.: An object-oriented approach to scene reconstruction. In: 1996 IEEE International Conference on Systems, Man and Cybernetics. Information Intelligence and Systems (Cat. No. 96CH35929). vol. 4, pp. 2475–2480. IEEE (1996) [3](#)
27. Huang, K., Wang, Y., Zhou, Z., Ding, T., Gao, S., Ma, Y.: Learning to parse wireframes in images of man-made environments. In: CVPR (2018) [2](#)
28. Jeong, Y., Nister, D., Steedly, D., Szeliski, R., Kweon, I.S.: Pushing the envelope of modern methods for bundle adjustment. *IEEE transactions on pattern analysis and machine intelligence* **34**(8), 1605–1617 (2011) [3](#)
29. Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J., Dellaert, F.: isam2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering. In: ICRA. IEEE (2011) [3](#)
30. Ke, Z., Tan, B., Xia, G.S., Shen, Y., Xue, N.: Interacted planes reveal 3d line mapping. *arXiv preprint arXiv:2602.01296* (2026) [4](#)
31. Klein, G., Murray, D.: Parallel tracking and mapping for small ar workspaces. In: 2007 6th IEEE and ACM international symposium on mixed and augmented reality. pp. 225–234. IEEE (2007) [3](#)
32. Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: g 2 o: A general framework for graph optimization. In: ICRA. IEEE (2011) [3](#)
33. Laidlow, T., Davison, A.J.: Simultaneous localisation and mapping with quadric surfaces. In: 2022 International Conference on 3D Vision (3DV). pp. 1–9. IEEE (2022) [4](#)
34. Li, H., Zhao, J., Bazin, J.C., Kim, P., Joo, K., Zhao, Z., Liu, Y.H.: Hong kong world: Leveraging structural regularity for line-based slam. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(11), 13035–13053 (2023) [4](#)
35. Li, X., He, Y., Lin, J., Liu, X.: Leveraging planar regularities for point line visual-inertial odometry. In: IROS. IEEE (2020) [2](#), [4](#), [6](#)
36. Liao, Z., Wang, W., Qi, X., Zhang, X., Xue, L., Jiao, J., Wei, R.: Object-oriented slam using quadrics and symmetry properties for indoor environments. *arXiv preprint arXiv:2004.05303* (2020) [4](#)
37. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647* (2025) [14](#)
38. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: Lightglue: Local feature matching at light speed. In: ICCV (2023) [11](#), [16](#)
39. Liu, S., Gao, Y., Zhang, T., Pautrat, R., Schönberger, J.L., Larsson, V., Pollefeys, M.: Robust incremental structure-from-motion with hybrid features. In: ECCV (2024) [4](#), [13](#), [14](#), [17](#)
40. Liu, S., Yu, Y., Pautrat, R., Pollefeys, M., Larsson, V.: 3d line mapping revisited. In: CVPR (2023) [4](#), [9](#), [11](#), [17](#)
41. Lourakis, M.I., Argyros, A.A.: Sba: A software package for generic sparse bundle adjustment. *ACM Transactions on Mathematical Software (TOMS)* **36**(1), 1–30 (2009) [3](#)
42. Ma, J., Wang, X., He, Y., Mei, X., Zhao, J.: Line-based stereo slam by junction matching and vanishing point alignment. *IEEE Access* **7** (2019) [4](#)
43. Ma, L., Kerl, C., Stückler, J., Cremers, D.: Cpa-slam: Consistent plane-model alignment for direct rgb-d slam. In: ICRA (2016) [4](#)
44. Moulon, P., Monasse, P., Perrot, R., Marlet, R.: OpenMVG: Open multiple view geometry. In: International Workshop on Reproducible Research in Pattern Recognition (2016) [3](#)

45. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: Orb-slam: a versatile and accurate monocular slam system. *IEEE Transactions on Robotics* **31**(5), 1147–1163 (2015) [3](#)
46. Nicholson, L., Milford, M., Sünderhauf, N.: Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam. *IEEE Robotics and Automation Letters* **4**(1), 1–8 (2018) [4](#)
47. Pan, L., Baráth, D., Pollefeys, M., Schönberger, J.L.: Global structure-from-motion revisited. In: *ECCV* (2024) [11](#)
48. Pautrat, R., Suárez, I., Yu, Y., Pollefeys, M., Larsson, V.: Gluestick: Robust image matching by sticking points and lines together. In: *ICCV* (2023) [4](#), [11](#), [16](#)
49. Pautrat, R., Barath, D., Larsson, V., Oswald, M.R., Pollefeys, M.: Deeplsd: Line segment detection and refinement with deep image gradients. In: *CVPR* (2023) [11](#), [16](#)
50. Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: *ICCV* (2021) [13](#), [14](#), [20](#)
51. Rother, C.: A new approach to vanishing point detection in architectural environments. *Image and Vision Computing* **20**(9-10), 647–655 (2002) [2](#), [4](#)
52. Schindler, G., Dellaert, F.: Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In: *CVPR* (2004) [4](#), [21](#)
53. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: *CVPR* (2016) [1](#), [3](#), [12](#), [17](#), [18](#)
54. Schops, T., Schonberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A multi-view stereo benchmark with high-resolution images and multi-camera videos. In: *CVPR* (2017) [13](#), [14](#)
55. Shotton, J., Glocker, B., Zach, C., Izadi, S., Criminisi, A., Fitzgibbon, A.: Scene coordinate regression forests for camera relocalization in RGB-D images. In: *CVPR* (2013) [14](#)
56. Shu, F., Wang, J., Pagani, A., Stricker, D.: Structure plp-slam: Efficient sparse mapping and localization using point, line and plane for monocular, rgb-d and stereo cameras. In: *ICRA* (2023) [4](#)
57. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: *ACM SIGGRAPH* (2006) [3](#)
58. Taguchi, Y., Jian, Y.D., Ramalingam, S., Feng, C.: Point-plane slam for hand-held 3d sensors. In: 2013 IEEE international conference on robotics and automation. pp. 5182–5189. *IEEE* (2013) [2](#), [4](#), [6](#)
59. Tang, C., Tan, P.: Ba-net: Dense bundle adjustment network. In: *International Conference on Learning Representations (ICLR)* (2019) [3](#)
60. Teed, Z., Deng, J.: Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *neurips* **34** (2021) [3](#)
61. Toldo, R., Fusiello, A.: Robust multiple structures estimation with j-linkage. In: *ECCV* (2008) [4](#), [11](#), [16](#)
62. Triggs, B.: Autocalibration from planar scenes. In: *ECCV* (1998) [4](#)
63. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings* (2000) [3](#), [5](#), [9](#)
64. Ubungazhibov, A., Pautrat, R., Suárez, I., Liu, S., Pollefeys, M., Larsson, V.: Light-gluestick: a fast and robust glue for joint point-line matching. In: *Computer Vision and Pattern Recognition Workshops (CVPRW)* (2025) [4](#)

65. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: CVPR (2025) [3](#)
66. Wang, J., Chen, M., Zhang, S., Karaev, N., Schönberger, J., Labatut, P., Bojanowski, P., Novotny, D., Vedaldi, A., Rupprecht, C.: Vggt- ω . In: CVPR (2026) [3](#)
67. Wang, J., Karaev, N., Rupprecht, C., Novotny, D.: Vggsfm: Visual geometry grounded deep structure from motion. In: CVPR (2024) [3](#)
68. Wang, R., Xu, S., Dai, C., Xiang, J., Deng, Y., Tong, X., Yang, J.: Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In: CVPR (2025) [11](#), [16](#)
69. Wang, R., Xu, S., Dong, Y., Deng, Y., Xiang, J., Lv, Z., Sun, G., Tong, X., Yang, J.: Moge-2: Accurate monocular geometry with metric scale and sharp details. In: arXiv (2025) [11](#), [16](#)
70. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3 : Permutation-equivariant visual geometry learning. arXiv preprint arXiv:2507.13347 (2025) [3](#)
71. Weber, S., Demmel, N., Chan, T.C., Cremers, D.: Power bundle adjustment for large-scale 3d reconstruction. In: CVPR (2023) [3](#)
72. Wilson, K., Snavely, N.: Robust global translations with ldsfm. In: ECCV (2014) [12](#), [13](#)
73. Wu, C.: Visualsfm: A visual structure from motion system. <http://www.cs.washington.edu/homes/ccwu/vsfm> (2011) [1](#), [3](#)
74. Wu, C., Agarwal, S., Curless, B., Seitz, S.M.: Multicore bundle adjustment. In: CVPR. IEEE (2011) [3](#)
75. Wu, Y., Zhang, Y., Zhu, D., Deng, Z., Sun, W., Chen, X., Zhang, J.: An object slam framework for association, mapping, and high-level tasks. IEEE Transactions on Robotics **39**(4), 2912–2932 (2023) [4](#)
76. Xu, K., Hao, Y., Yuan, S., Wang, C., Xie, L.: Airslam: An efficient and illumination-robust point-line visual slam system. IEEE Transactions on Robotics **41**, 1673–1692 (2025) [4](#)
77. Xue, N., Wu, T., Bai, S., Wang, F., Xia, G.S., Zhang, L., Torr, P.H.: Holistically-attracted wireframe parsing. In: CVPR (2020) [2](#), [4](#)
78. Yang, S., Scherer, S.: Cubeslam: Monocular 3-d object slam. IEEE Transactions on Robotics **35**(4), 925–938 (2019) [4](#)
79. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: ICCV (2023) [14](#), [20](#)
80. Zach, C.: Robust bundle adjustment revisited. In: ECCV (2014) [3](#)
81. Zhang, Y., Keetha, N., Lyu, C., Jhamb, B., Chen, Y., Qiu, Y., Karhade, J., Jha, S., Hu, Y., Ramanan, D., Scherer, S., Wang, W.: Ufm: A simple path towards unified dense correspondence with flow. In: arXiv (2025) [11](#)
82. Zhao, X., Wu, X., Chen, W., Chen, P.C., Xu, Q., Li, Z.: Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. IEEE Transactions on Instrumentation and Measurement **72**, 1–16 (2023) [11](#), [16](#)
83. Zhou, L., Liu, J., Zhai, F., Ai, P., Ren, K., Mao, Y., Huang, G., Meng, Z., Kaess, M.: Efficient bundle adjustment for coplanar points and lines. In: ICRA. IEEE (2023) [4](#)
84. Zhou, L., Wang, S., Kaess, M.: A fast and accurate solution for pose estimation from 3d correspondences. In: ICRA (2020) [4](#)
85. Zhou, Y., Qi, H., Zhai, Y., Sun, Q., Chen, Z., Wei, L.Y., Ma, Y.: Learning to reconstruct 3d manhattan wireframes from a single image. In: ICCV (2019) [2](#), [4](#)

- 86. Zhuang, B., Cheong, L.F., Lee, G.H.: Baseline desensitizing in translation averaging. In: CVPR (2018) [11](#)
- 87. Zuo, X., Xie, X., Liu, Y., Huang, G.: Robust visual slam with point and line features. In: IROS (2017) [4](#)