# COLMAP 3D重建管道：工作流程、原理与实现方案专家报告

## Section 1: COLMAP架构与3D重建流程概述

### 1.1. COLMAP在3D重建领域的定位与核心功能

COLMAP是一个通用且全面的图像级三维重建软件，它集成了Structure-from-Motion (SfM) 和 Multi-View Stereo (MVS) 两个核心管道，提供了端到端（end-to-end）的重建能力 1。该软件能够处理有序（如视频帧）和无序（如互联网照片）的图像集合，旨在通过结合高精度算法和高效实现，生成高质量的稀疏点云、相机姿态和最终的密集几何模型 1。由于其输出精度极高，COLMAP所估计的相机姿态和场景几何常常被计算机视觉社区视为“黄金标准”或“地面真值”，尤其在作为新兴技术（如NeRF）的姿态初始化工具时，发挥着至关重要的作用 33。COLMAP采用BSD许可，确保了其在学术研究和工业应用中的广泛使用 1。

### 1.2. COLMAP的核心理念：增量式SfM与MVS的集成

COLMAP的重建策略基于经典的\*\*增量式SfM (Incremental SfM)\*\*范式 3。与全局式SfM一次性求解所有相机姿态不同，增量式方法首先从一个高质量、高置信度的初始图像对（“种子”）开始重建，随后以渐进的方式将新的图像注册到现有模型中，并在每一步执行局部精化，直至收敛 33。这种策略的优势在于其固有的鲁棒性，能够有效应对初始估计误差，并通过迭代优化确保整个重建过程的质量稳定 5。

完整的COLMAP流程可以清晰地分为两个主要阶段 1：

1. **SfM（稀疏重建）:** 负责估计相机的内外参、姿态以及场景的稀疏三维结构（稀疏点云）。
2. **MVS（密集重建）:** 利用SfM阶段计算出的精确相机参数，通过多视角几何方法恢复场景的密集几何细节，生成深度图、法线图和最终的密集点云 6。

### 1.3. 完整的命令行工作流程（CLI）概览与指令映射

COLMAP提供了图形用户界面（GUI）和命令行接口（CLI）8。对于自动化和大规模处理，CLI是标准的选择。CLI将整个重建流程分解为一系列独立的、按顺序执行的模块化命令 8，这不仅是为了简化脚本编写，更是为了提供卓越的技术灵活性和控制能力。这种模块化设计允许用户在流程的任何环节进行干预、调试或替换组件，并且可以在高性能计算环境中并行化资源密集型步骤，如MVS阶段。

以下表格总结了COLMAP端到端标准命令行工作流程中的关键步骤、实现命令和核心功能：

Table 1: COLMAP端到端标准命令行工作流程

| **阶段 (Pipeline Stage)** | **CLI 命令 (Implementation)** | **原理功能 (Principle)** | **关键输出/文件** |
| --- | --- | --- | --- |
| **特征提取 (Feature Extraction)** | colmap feature\_extractor | 识别兴趣点并计算描述符 (e.g., SIFT) | database.db (存储关键点/描述符) 8 |
| **特征匹配 (Feature Matching)** | colmap exhaustive\_matcher | 找到所有图像对间的原始匹配，并进行比率测试 | database.db (存储原始匹配) 8 |
| **稀疏映射 (Sparse Mapping)** | colmap mapper | 执行增量式SfM：几何验证 (RANSAC)、三角测量、BA优化 | sparse/0/ (相机姿态, 稀疏点云) 8 |
| **图像去畸变 (Undistortion)** | colmap image\_undistorter | 矫正图像和相机参数，准备MVS输入 | dense/ (图像, 相机参数) 8 |
| **PatchMatch 立体匹配 (MVS)** | colmap patch\_match\_stereo | 逐视图估计深度图和法线图，利用多视角几何一致性 | dense/stereo/depth\_maps/ (深度图) 8 |
| **深度图融合 (Fusion)** | colmap stereo\_fusion | 聚合所有深度图和法线图，生成最终密集点云 | dense/fused.ply (密集点云) |

## Section 2: 相机与输入数据预处理

### 2.1. 相机模型的类型与选择

准确建模相机内参是SfM成功的先决条件。COLMAP支持多种相机模型，以适应不同的数据采集设备和畸变特征 9。

* **基本模型:** SIMPLE\_PINHOLE 和 PINHOLE 适用于图像已提前进行几何矫正的情况 9。
* **径向畸变模型:** SIMPLE\_RADIAL 和 RADIAL 是最常用的模型，尤其推荐用于内参未知且每张图像可能具有不同校准的场景（例如互联网照片）99。这些模型是OpenCV模型简化版，仅对径向畸变进行建模 9。
* **鱼眼模型:** SIMPLE\_RADIAL\_FISHEYE, RADIAL\_FISHEYE, OPENCV\_FISHEYE 等专门用于建模鱼眼镜头产生的严重畸变效应 99。

在 feature\_extractor 步骤中，用户可以指定相机模型。COLMAP能够自动从图像的EXIF元数据中提取焦距信息作为内参的良好初始化值 6。

### 2.2. 数据存储结构与数据库管理

COLMAP在数据处理过程中采用了结构化的文件格式进行存储和管理。

* **数据库文件:** 所有的输入信息、关键点、描述符和原始匹配对都存储在一个SQLite数据库文件（database.db）中 4。这是SfM前端（提取和匹配）的核心数据结构。
* **稀疏模型输出格式:** 稀疏重建结果（相机姿态和稀疏点云）默认存储在 sparse/ 目录下，并支持两种格式：机器可读且快速的**二进制格式（.bin）和人类可读的文本格式（.txt）** 1010。这些信息被拆分为多个文件，包括 cameras.txt、images.txt 和 points3D.txt 等 1010。

### 2.3. 共享内参的重要性与系统约束

在SfM管道中，用户可以选择让COLMAP自动估计相机内参。但是，如果每个图像都拥有独立的内参，自动估计过程往往会失败 99。

这种现象反映了非线性优化中的一个基本约束：在没有足够数据冗余的情况下，焦距/主点等内参与平移/深度等外参之间存在高度相关性（高协方差），使得优化系统处于欠约束状态。因此，COLMAP在设计上通过允许**多张图像共享一个相机内参ID** (CAMERA\_ID) 1010 来解决这一问题。当内参被多张图像共享时，优化过程获得了充分的冗余观测，从而能够更准确和稳健地精化焦距、畸变参数 9。

## Section 3: 特征工程：提取与匹配的原理与实现

### 3.1. 特征提取原理：SIFT的优化与替代方案

COLMAP重建流程的起始步骤是特征检测和提取，通常通过 colmap feature\_extractor 命令执行 8。

* **核心算法:** COLMAP传统上依赖SIFT（尺度不变特征变换）算法或其变体 11。SIFT 描述符因其在尺度和旋转变化下的鲁棒性，长期以来一直是SfM管道的基石 12。
* **性能增强:** 为了应对更具挑战性的数据集（如图像存在较大视角变化或失真），COLMAP推荐使用优化方案来增强特征的几何不变性 1212：
  1. **仿射特征形状估计 (--SiftExtraction.estimate\_affine\_shape=true)**: 允许特征适应非尺度和旋转的仿射失真 1212。
  2. **域尺寸池化 (DSP-SIFT, --SiftExtraction.domain\_size\_pooling=true)**: 提高描述符在深度不连续区域和视角变化时的稳健性 1212。
* **自定义集成:** COLMAP的数据库驱动结构允许高级用户绕过内置的提取器，手动创建并导入外部特征（如SuperPoint或其他深度学习特征）到 database.db 中，随后继续运行COLMAP的匹配和映射步骤，这为研究人员提供了极大的灵活性 11。

### 3.2. 特征匹配与几何验证的协同工作

特征匹配旨在识别在不同图像中观测到的同一场景点的2D对应关系。

* **匹配策略:** COLMAP支持多种匹配命令，包括 exhaustive\_matcher（适用于所有图像对，适合小型数据集）和 sequential\_matcher（适用于有序图像序列，如视频帧）88。此外，用户还可以启用**引导匹配 (--SiftMatching.guided\_matching=true)** 来进一步提高匹配的质量和数量 1212。
* **几何验证（RANSAC）:** 原始匹配集通常包含大量的错误对应关系（外点）。SfM的健壮性取决于消除这些外点。几何验证基于对极几何约束，通过**RANSAC (Random Sample Consensus)** 算法估计基本矩阵或本质矩阵，以剔除不符合几何约束的匹配对 13。

### 3.3. RANSAC几何验证的实现与参数调优

几何验证基于对极几何约束，通过**RANSAC (Random Sample Consensus)** 算法估计基本矩阵或本质矩阵 13。RANSAC 是一个解决鲁棒模型估计问题的迭代算法，它遵循“假设-验证”的框架，能够从包含大量噪声和外点（Outliers）的测量数据中稳健地估计模型参数 1414。 RANSAC采用“假设-验证”框架，迭代地从最小样本集中估计模型，并评估其内点数量，直到满足统计置信度 14。

几何验证的鲁棒性由一组关键参数控制 1515：

* **最大对极误差 (TwoViewGeometry.max\_error):** 默认值为4.0像素，定义了匹配点作为内点所允许的最大重投影误差 15。
* **置信度 (TwoViewGeometry.confidence):** 默认值0.999，用于计算所需的最大RANSAC迭代次数 15。
* **最小内点比率 (TwoViewGeometry.min\_inlier\_ratio):** 默认值0.25。这是系统假设的先验最小内点比例，用于限制RANSAC的最大迭代次数 15。

特征提取的质量与几何验证的计算成本之间存在直接的因果关系。在RANSAC理论中，先验内点比率的提高将急剧减少实现目标置信度所需的迭代次数 14。这意味着在特征工程阶段投入的计算资源，可以直接转化为几何验证阶段更低的计算消耗和更高的鲁棒性。

## Section 4: 稀疏重建核心：增量式SfM

### 4.1. 增量式SfM流程与模型初始化

稀疏重建由 colmap mapper 命令执行，采用增量式方法逐步构建场景 8。这种增量式 SfM 方法通过逐步注册新图像和精化重建来创建高质量的种子模型，并最终收敛 3。

* **初始化:** 首先，系统必须选择一个高质量的“初始图像对”，该图像对必须具有足够的几何验证内点，并能通过对极几何准确估计相对姿态 13。随后，通过\*\*三角测量（Triangulation）\*\*根据该相对姿态和2D匹配点生成初始的3D点云 13。
* **三角测量原理:** 3D点的世界坐标是通过多张图像中观测到的2D射线在空间中的交汇点确定的 13。COLMAP采用鲁棒的三角测量方法，通常结合RANSAC进行递归采样和调整采样区域，以确保生成的3D点具有较高的精度和抗噪能力 1313。

### 4.2. 模型的扩展与图像注册

一旦建立了初始模型，增量式SfM进入模型扩展阶段：

* **图像注册:** 对于每一张新的未注册图像，系统利用已知的3D点云和该图像中的2D特征观测，通过**PnP (Perspective-n-Point)** 算法来估计其相机位姿（外部参数：旋转R和翻译T）13。
* **鲁棒性:** 姿态估计过程中再次引入RANSAC，以确保PnP计算对2D-3D匹配中的外点具有鲁棒性 13。只有当图像的姿态被稳健地估计并具有足够数量的内点时，该图像才会被正式注册到稀疏模型中，并随后用于三角测量新的3D点 6。

### 4.3. 稀疏重建数据格式解析与几何约定

稀疏重建结果存储在 sparse/ 目录下的文本文件（如 images.txt 和 points3D.txt）中，这些文件是后续MVS和其他下游应用的基础 10。

#### 姿态表示 (images.txt)

images.txt 文件以两行数据存储每张图像的信息。第一行包含图像的姿态和元数据：

* **旋转:** 使用四元数 $(QW, QX, QY, QZ)$ 定义，遵循Hamilton约定，表示从世界坐标系到相机坐标系的旋转 1010。
* **平移:** 使用欧氏平移向量 $(TX, TY, TZ)$ 定义 10。
* **相机坐标系约定:** COLMAP定义局部相机坐标系为X轴向右、Y轴向下、Z轴向前 1010。
* **相机中心计算:** 世界坐标系下的相机中心 $C$ 是通过外参转换计算的，其公式为 $C = -R^t \cdot T$，其中 $R^t$ 是由四元数组成的旋转矩阵的转置（即逆矩阵）1010。

#### 3D点结构 (points3D.txt)

points3D.txt 记录了场景中所有三角测量得到的3D点 10。

* **核心信息:** 包含世界坐标 $(X, Y, Z)$、颜色 $(R, G, B)$ 和关键的**重投影误差 (ERROR)** 1010。重投影误差以像素为单位，是评估单个3D点精度和几何一致性的主要指标，仅在全局Bundle Adjustment后更新 1010。
* **观测轨道 (TRACK):** 每个3D点都记录了一个轨道列表，详细列出观测到该点的所有图像ID及其对应的2D关键点索引 1010。

## Section 5: 几何精化核心：Bundle Adjustment (BA)

### 5.1. Bundle Adjustment (BA) 的理论基础

Bundle Adjustment (BA) 是SfM管道的核心几何精化技术 3，被誉为重建精度的“黄金标准” 3。BA是一种**非线性最小二乘优化**问题 1616，目标是**联合优化**所有相机参数（内参、外参）和所有3D点坐标 1616，以最小化所有2D观测点与其对应的3D点在图像平面上的投影点之间的总重投影误差 16。

该优化问题通常使用**Levenberg-Marquardt (LM) 算法**求解 16。BA的成功高度依赖于良好的初始化结果，即初始估计的相机姿态和3D点位置必须足够接近真实解 33。

### 5.2. BA的迭代优化策略 (局部与全局)

为了平衡重建的实时性和全局精度，COLMAP采用了局部和全局相结合的迭代BA策略 5：

* **局部 BA (Local BA):** 在每次成功注册新图像后，系统会执行局部BA。它仅优化新注册图像的姿态、所有新三角测量的3D点，以及与这些新3D点共享观测的其他图像的姿态 521。默认情况下，局部BA的最大迭代次数设置为25次 (ba\_local\_max\_num\_iterations) 1717。
* **全局 BA (Global BA):** 全局BA的计算成本更高，但能确保整个模型的几何一致性。它定期触发，通常在注册的图像数量达到一定增长率（默认增长率为1.1，即 ba\_global\_frames\_ratio = 1.1）时执行 1717。全局BA会**联合优化当前模型中所有已注册图像的姿态和所有3D点坐标** 513。
* **参数固定与精化:** COLMAP默认会在重建过程中精化焦距和畸变参数，但通常保持主点（Principal Point）不变 12。仅在所有图像注册完成后，才建议在最终的全局BA中尝试精化主点 1212。

### 5.3. 基于Ceres Solver的BA实现方案与高性能计算

COLMAP利用Google开发的**Ceres Solver**作为其非线性优化引擎 17。Ceres Solver是专门为处理大规模、稀疏的最小二乘问题（如BA）设计的，其高效性对于SfM的性能至关重要 16。

对于处理超大规模数据集，BA往往成为SfM管道中最耗时的步骤。COLMAP支持利用GPU加速Bundle Adjustment的求解过程，特别是稀疏矩阵的求解 18。然而，启用GPU加速的BA并非易事，它要求用户进行定制化的编译流程：必须下载和配置CUDA和cuDSS（NVIDIA提供的求解器库），并手动修改如Ceres和COLMAP相关的CMake/vcpkg配置文件，以确保Ceres在编译时启用了CUDA支持 1818。

这种对专业编译流程和特定硬件（如cuDSS）的依赖，揭示了一个技术事实：虽然SfM的前端部分（特征提取、RANSAC）通常被认为是CPU密集型任务，但当面对大型数据集时，最终的几何精化瓶颈会转移到**BA求解器的并行化能力**上。

## Section 6: 密集重建（MVS）的原理与实践

### 6.1. MVS预处理：图像去畸变

在SfM阶段生成精确的稀疏模型后，下一步是利用多视角立体匹配（MVS）生成高密度的几何数据。MVS算法通常要求输入图像是无畸变的 6。

* **CLI 命令:** colmap image\_undistorter 8
* **原理与实现:** 此命令利用SfM阶段在 cameras.txt 中估计出的相机内参和畸变系数，对原始图像进行几何矫正，并输出新的、无畸变的图像集和相应的投影参数 8。矫正后的数据存储在 dense/ 目录下 8。

### 6.2. PatchMatch Multi-View Stereo (PMS) 算法

COLMAP的MVS管道基于高效且鲁棒的**PatchMatch Stereo (PMS)** 算法，用于逐视图地估计深度图和法线图 77。

* **算法核心:** PMS是一种基于随机优化的迭代方法 7：
  1. **初始化:** 为每个像素随机初始化深度和法线假设。
  2. **传播 (Propagation):** 将邻域像素的深度/法线假设传播到当前像素，利用场景的局部平滑性。
  3. **随机精化 (Random Refinement):** 在当前假设周围的深度和法线空间中随机采样，寻找光度一致性更优的匹配。
  4. **几何一致性检查:** 通过多视角几何约束来验证假设的质量，以确保深度估计的准确性 7。
* **性能考量与内存管理:** PMS是典型的GPU密集型任务 1212。当处理高分辨率图像或大型场景时，PatchMatch立体匹配容易耗尽GPU内存 1212。为了缓解内存压力，用户可以调整参数，例如减小最大图像尺寸 (--PatchMatchStereo.max\_image\_size) 或减少用于匹配的源视图数量，但这可能会牺牲密集重建的分辨率和质量 121212。

### 6.3. 深度图与法线图融合 (Stereo Fusion)

在PMS阶段为每个视图生成深度图和法线图后，必须通过融合过程将这些数据聚合为一个统一的密集点云。

* **CLI 命令:** colmap stereo\_fusion
* **原理:** 融合阶段将每个参考图像的像素投影到3D世界坐标系中，并通过多视图一致性检查来过滤不准确的3D点 19。 **Point Cloud Generation:** 每个像素被投影到3D世界坐标系中，并由所有邻近视图进行观测 19。只有那些在多个邻域视图中保持高度几何和光度一致性的点才会被保留 1919。
* **输出格式:** 最终结果是一个高密度的点云文件，通常以.ply格式输出，包含精确的几何坐标和颜色信息 19。

## Section 7: 技术细节、数据格式与性能分析

### 7.1. 稀疏重建模型输出格式详解

COLMAP稀疏模型（存储在 sparse/ 目录下）的结构化输出是其作为开放平台的重要特征 10。这些文本文件详细记录了重建的所有几何和拓扑信息 10。

Table 2: 稀疏重建模型关键输出文件字段

| **文件名称** | **核心信息** | **关键字段** | **原理意义** |
| --- | --- | --- | --- |
| **cameras.txt** | 相机内参 | CAMERA\_ID, MODEL, WIDTH, HEIGHT, PARAMS | 定义相机的几何投影模型和畸变系数 [9, 10]10。 |
| **images.txt (Line 1)** | 图像位姿 | QW, QX, QY, QZ, TX, TY, TZ, CAMERA\_ID, NAME | 以外参（R, T）形式存储图像在世界坐标系中的位置和方向 1010。 |
| **images.txt (Line 2)** | 2D观测 | POINTS2D as (X, Y, POINT3D\_ID) | 关键点的像素坐标及其与3D点的对应关系；$-1$ 表示未三角测量的关键点 1010。 |
| **points3D.txt** | 3D点信息 | X, Y, Z, R, G, B, ERROR, TRACK | 3D点的世界坐标、颜色、重投影误差（精度指标）以及所有观测此点的图像轨道 (Track) 列表 1010。 |

### 7.2. COLMAP在研究和评估中的作用

COLMAP的模块化设计和详尽的数据输出使其超越了单纯的重建工具，成为一个重要的诊断和评估框架 22。

首先，points3D.txt 中精确记录的**重投影误差 (ERROR)** 为研究人员提供了一个量化重建精度的核心指标 1010。其次，每个3D点的完整观测轨道列表（TRACK）提供了场景几何和图像连通性的细致拓扑结构。

这种透明且可追踪的数据结构，使得研究人员可以轻松地验证新型算法的几何一致性和准确性。例如，通过运行 feature\_extractor 和 matcher 后，研究人员可以将自己开发的几何算法的匹配结果注入到 database.db 中 11，然后继续运行 mapper，并利用最终的重投影误差和相机姿态来客观评估其新算法的性能 22。

### 7.3. 性能考量与优化总结

性能分析显示，COLMAP的两个主要阶段具有不同的硬件瓶颈 23：

1. **SfM阶段（特征提取、RANSAC、BA）:** 总体上是CPU密集型任务，受益于多核处理能力 23。
2. **MVS阶段（PatchMatch Stereo）:** 是GPU密集型任务，性能受限于GPU的内存容量和计算速度 12。

为了优化性能：

* **SfM优化:** 确保高质量的特征提取（如DSP-SIFT）以减少RANSAC迭代次数，并在处理超大规模模型时，投入资源进行复杂的GPU加速BA编译 1218。
* **MVS优化:** 在内存成为瓶颈时，虽然可以选择切换到CPU匹配或降低匹配数，但最直接的MVS性能权衡是减少图像处理尺寸 (--PatchMatchStereo.max\_image\_size)，但这需要接受最终密集点云分辨率的下降 121212。

## 结论

COLMAP提供了一个高度结构化、鲁棒且灵活的3D重建流程。其核心在于增量式SfM范式，该范式通过前端的SIFT/RANSAC几何验证保证数据质量，并通过后端的Ceres Bundle Adjustment进行精确的联合优化。随后，MVS管道利用PatchMatch Stereo生成密集几何。该系统的模块化设计和详尽的输出格式（尤其是对重投影误差和相机姿态的精确记录）使其成为业界和学术界不可或缺的基准工具 3。对于大规模应用，性能优化必须针对不同阶段的资源瓶颈进行调整：SfM阶段关注BA求解器的加速集成（GPU/cuDSS），而MVS阶段则必须平衡GPU内存限制与最终模型的精度需求。

#### 引用的著作

1. COLMAP - Structure-from-Motion and Multi-View Stereo - GitHub, 访问时间为 十一月 4, 2025， <https://github.com/colmap/colmap>
2. COLMAP - Structure-from-Motion and Multi-View Stereo - Johannes Schönberger, 访问时间为 十一月 4, 2025， <https://demuc.de/colmap/>
3. Scene Coordinate Reconstruction: Posing of Image Collections via Incremental Learning of a Relocalizer - arXiv, 访问时间为 十一月 4, 2025， <https://arxiv.org/html/2404.14351v1>
4. Building Better Models: Benchmarking Feature Extraction and Matching for Structure from Motion at Construction Sites - MDPI, 访问时间为 十一月 4, 2025， <https://www.mdpi.com/2072-4292/16/16/2974>
5. How COLMAP Implements Incremental SfM with Bundle Adjustment - Patsnap Eureka, 访问时间为 十一月 4, 2025， <https://eureka.patsnap.com/article/how-colmap-implements-incremental-sfm-with-bundle-adjustment>
6. Tutorial — COLMAP 3.13.0.dev0 | a5332f46 (2025-07-05) documentation, 访问时间为 十一月 4, 2025， <https://colmap.github.io/tutorial.html>
7. PatchMatch Multi-View Stereo. Understand how the MVS tool provided by… | by Thomas Rouch | Better Programming, 访问时间为 十一月 4, 2025， <https://betterprogramming.pub/patchmatch-multi-view-stereo-1-2-fc46e5dfe912>
8. Command-line Interface — COLMAP 3.13.0.dev0 | a5332f46 (2025 ..., 访问时间为 十一月 4, 2025， <https://colmap.github.io/cli.html>
9. Camera Models — COLMAP 3.13.0.dev0 | a5332f46 (2025-07-05) documentation, 访问时间为 十一月 4, 2025， <https://colmap.github.io/cameras.html>
10. Output Format — COLMAP 3.13.0.dev0 | a5332f46 (2025-07-05 ..., 访问时间为 十一月 4, 2025， <https://colmap.github.io/format.html>
11. Using different feature extractor in colmap : r/GaussianSplatting - Reddit, 访问时间为 十一月 4, 2025， <https://www.reddit.com/r/GaussianSplatting/comments/1ib75c2/using_different_feature_extractor_in_colmap/>
12. Frequently Asked Questions — COLMAP 3.13.0.dev0 | a5332f46 ..., 访问时间为 十一月 4, 2025， <https://colmap.github.io/faq.html>
13. Colmap-PCD: An Open-source Tool for Fine Image-to-point cloud Registration - arXiv, 访问时间为 十一月 4, 2025， <https://arxiv.org/html/2310.05504>
14. Fixing the RANSAC Stopping Criterion - arXiv, 访问时间为 十一月 4, 2025， <https://arxiv.org/html/2503.07829v1>
15. mwtarnowski/colmap-parameters - GitHub, 访问时间为 十一月 4, 2025， <https://github.com/mwtarnowski/colmap-parameters>
16. DEEPSFM: STRUCTURE FROM MOTION VIA DEEP BUNDLE ADJUSTMENT - OpenReview, 访问时间为 十一月 4, 2025， <https://openreview.net/pdf?id=SyeD0RVtvS>
17. pycolmap — COLMAP 3.13.0.dev0 | a5332f46 (2025-07-05) documentation, 访问时间为 十一月 4, 2025， <https://colmap.github.io/pycolmap/pycolmap.html>
18. Summarizing how to utilize the GPU for bundle adjustment for future users. #3163 - GitHub, 访问时间为 十一月 4, 2025， <https://github.com/colmap/colmap/issues/3163>
19. morsingher/depth\_fusion: A simple C++ algorithm for converting depth and normal maps to a point cloud - GitHub, 访问时间为 十一月 4, 2025， <https://github.com/morsingher/depth_fusion>
20. Geometric verification of manual matches - Google Groups, 访问时间为 十一月 4, 2025， <https://groups.google.com/g/colmap/c/I2k56QRi9Wg>
21. Robust Incremental Structure-from-Motion with Hybrid Features - Johannes Schönberger, 访问时间为 十一月 4, 2025， <https://demuc.de/papers/liu2024hybrid.pdf>
22. Global Structure-from-Motion Revisited - arXiv, 访问时间为 十一月 4, 2025， <https://arxiv.org/html/2407.20219v1>
23. Open Source 3D Reconstruction Showdown: Complete Tools Comparison (2025) | TripoSR AI, 访问时间为 十一月 4, 2025， <https://www.triposrai.com/posts/open-source-3d-reconstruction-showdown/>