

# Sampling-Based Path Planning for High-Quality Aerial 3D Reconstruction of Urban Scenes

Feihu Yan<sup>1</sup>, Enyong Xia<sup>1</sup>, Zhaoxin Li<sup>2</sup> and Zhong Zhou<sup>1,\*</sup>

## 摘要

无人机可以拍摄高质量的航拍照片，并已广泛用于大规模城市 3D 重建。然而，即使借助商业飞控软件，对于非专业用户来说，在复杂的城市环境中拍摄全覆盖航拍照片仍然是一项具有挑战性的任务，通常会导致 3D 重建不完整。在本文中，提出了一种新颖的路径规划方法，用于城市场景的高质量空中 3D 重建。所提出的方法首先捕获航拍照片，然后按照初始路径生成粗略的 3D 模型作为先验知识。然后，根据先验模型相应可见区域的完整性和准确性，生成并评估具有约束位置和方向的 3D 视点。最后，通过平滑连接最优视点产生最优路径。

**关键词：**无人机；路径规划；3D 重构；城市场景

## 1 引言

现在无人机凭借体积小、灵活性高等优势，被广泛应用于智能交通、工业检测、农业等各个领域。装载摄像头的无人机能够在复杂的城市环境中借助最先进的运动结构 (SfM) 和多视图立体 (MVS) 算法进行拍摄照片来进行三维重建。通常，这些高质量的 3D 模型对于许多高级应用至关重要，包括增强现实 (AR)、虚拟现实 (VR) 和机器人导航。基于无人机的 3D 场景扫描想要去完全满足城市场景的全覆盖是很难实现的，一方面需要有熟练无人机技术的操作人员对无人机进行操作采集城市场景，同时城市的复杂场景可能会因为一些不可控的因素导致安全事故的发生，另一方面，现有的商业飞行采集是通过在指定城市高空安全距离进行拍摄照片，虽然这样能提高无人机的安全性，但是对城市建筑场景重要信息的捕捉通常会有遗漏，导致场景重建不完整或者重建的质量不好。

## 2 相关工作

### 2.1 Aerial Reconstruction

近年来，航空重建已广泛应用于考古学<sup>[1]</sup>或智慧城市等多种应用领域。当前研究提出了各种与无人机机载传感器相关的解决方案，例如单目 RGB 相机、立体相机、RGB-D 传感器、多光谱相机或激光雷达。目前主流的航拍重建软件一般都是基于 SfM<sup>[2]</sup>，然而，传统的 SfM 方法需要数小时才能生成最终的重建模型；因此，这些方法不适合实时使用。相比之下，另一个旨在实时重建的选项是同时定位和映射 (SLAM)<sup>[3]</sup>，它能够同时输出环境的 3D 地图和相机姿势，一些研究人员利用 SLAM 进行实时空中重建<sup>[4][5]</sup>，但是传统的单目 SLAM 方法总是在跟踪上进行，只能得到稀疏或半稠密的地图。此外，与 SfM 方法相比，该方法生成的 3D 地图的精度要差很多。

### 2.2 Path Planning for Scene Reconstruction

目前，现有许多研究关于 3D 重建的视点选择方法。Kriegel 等人<sup>[6]</sup>提出了一种自主 3D 对象建模系统，该系统使用对象表面和体积模型搜索扫描路径以最大限度地提高 3D 模型的质量。Wu 等人<sup>[7]</sup>提出了一种针对复杂物理对象的自主高质量扫描方法，该方法基于泊松的重建质量测量来计算 NBV。

Xu 等人<sup>[8]</sup>结合基于 RGB-D 的 SLAM 系统，通过对象引导扫描和重建对室内场景进行自动扫描。这些提到的方法都实现了对物体或室内场景的高质量的重建；然而，由于计算复杂度高，这些方法很难应用于大规模的户外场景。最近研究工作有部分是集中在先探索后利用的方法上，在“探索”阶段，生成场景几何的粗略估计，可用于计算优化“利用”阶段的轨迹以实现高质量 3D 重建。Roberts 等人<sup>[9]</sup>提出了一个数学模型来估计摄像机轨迹对多视图立体重建的有用性。Peng 和 Isler<sup>[10]</sup>使用自适应查看矩形算法解决了视图规划问题；然而，他们的视图规划方法需要两次以上的迭代才能构建高质量的重建。Hepp 等人<sup>[11]</sup>采用分层体积占用图和子模块优化公式来选择最佳候选视点。Smith 等人<sup>[12]</sup>提出了一种可重构启发式方法来不断优化路径规划。对于这些方法，初始网格模型是从开源 SfM 和 MVS 方法或商业软件获得的，然而，错误的三角网格经常出现在初始模型中并干扰后续过程，通常是由散布在空间中的异常值和相邻 3D 点的错误连接引起的。

### 3 本文方法

#### 3.1 本文方法概述

此部分对本文将要复现的工作进行概述，图的插入如图 1 所示：

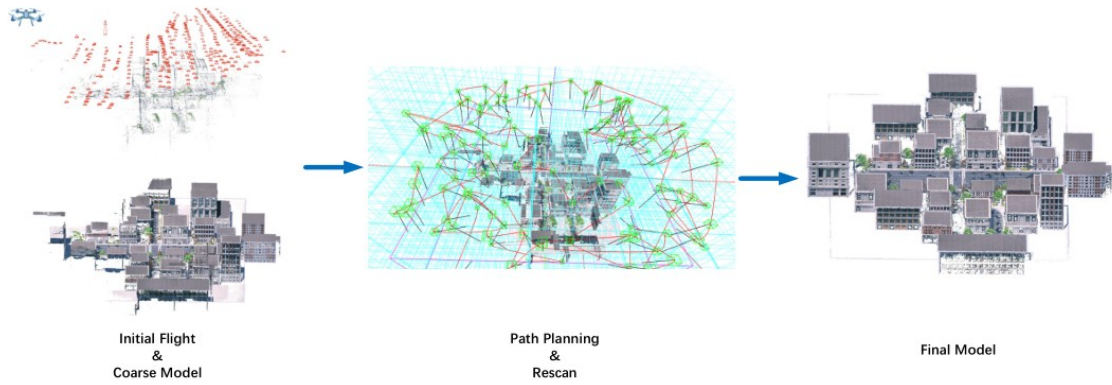


图 1: 方法示意图

#### 3.2 Prior Model Construction

##### 3.2.1 Coarse Reconstruction

本文方法的第一步包括初始的飞行和粗模型的重建，初始飞行不限于手动控制或者自动导航。由于首次飞行的捕获过程可能会非常的繁琐和消耗大量的计算资源，因此，需要提高粗模型重建的效率。受到快速 3D 重建方法<sup>[13]</sup>的启发，可以先为每个图像提取 ORB 特征<sup>[14]</sup>，然后，匹配当前图像和相邻关键图像的特征以评估重叠。一旦匹配的特征点数量小于阈值，则将当前图像用作新的关键图像，以确保所选图像保留足够的重叠。值得注意的是，直接丢弃特征点很少的图像。由于 ORB 特征的计算和匹配速度非常快<sup>[15]</sup>，关键图像的选择可以立即完成。

##### 3.2.2 Pre-Processing

论文的方法直接分析重建的密集点云，为了有效地定位不完整和低质量的区域，对数据进行预处理。首先，提取粗糙模型的边界框，计算出所有点的法线后，提取三个主导轴来构建边界框；第二步，进行泊松曲面重建，对初始模型中不完整的区域通过使用隐式泊松曲面计算进行覆盖；第三步，表面

采样，通过采样来降低计算的复杂度和评估重建质量，使用泊松圆盘的采样方法在模型表面进行均匀随机采样，得到一组有向的采样点。

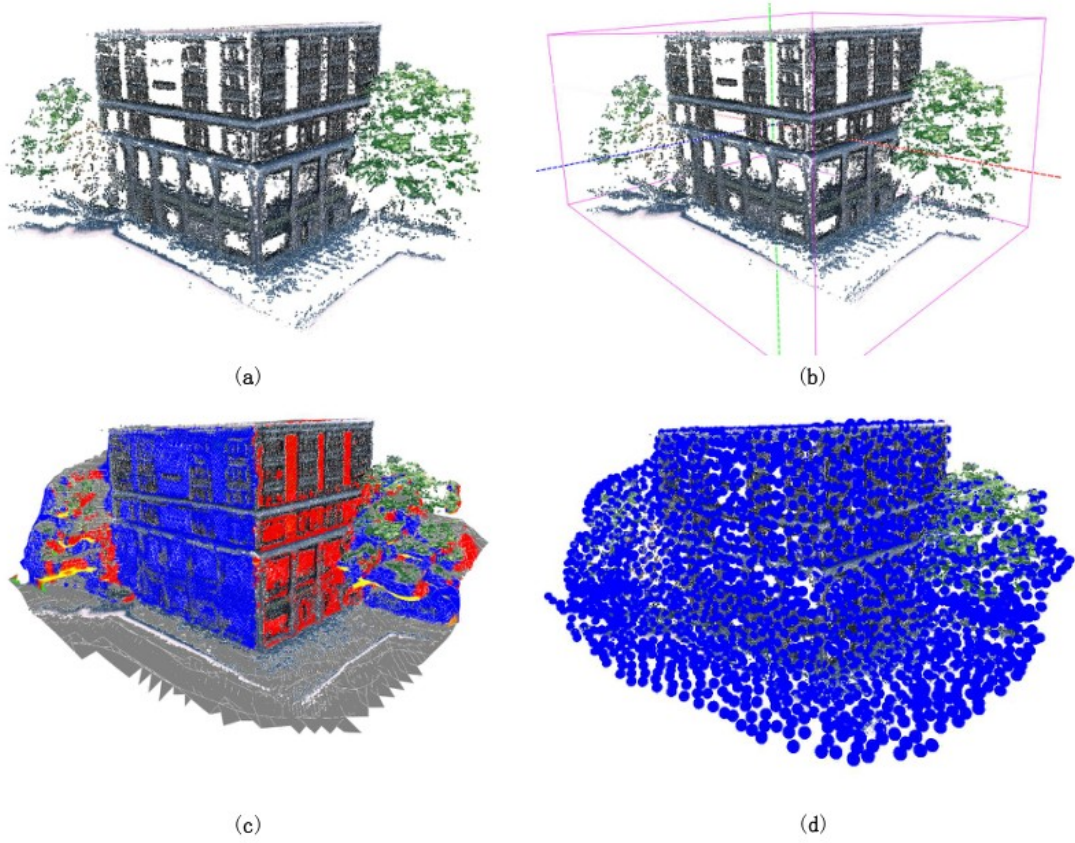


图 2: 预处理示意图

### 3.2.3 Quality Evaluation of the Initial Point Cloud

正如前面所提到的，隐式泊松曲面可以完全覆盖整个场景，并且采样点在等值面上有规律地分布，可以使用样本点的质量来估计粗模型的质量，如果局部点云足够准确和密集，则恢复的泊松曲面将非常准确，如果局部点云质量低或缺失区域，则泊松曲面也将缺少相应的几何细节。本文为每个粗模型表面上的样本点  $s_i$  计算一个完整性分数  $g_c(s_i)$  和一个平滑度分数  $g_s(s_i)$ 。

完整性分数。完整性分数  $g_c$  表示样本点  $s_i$  周围的初始点云中存在孔洞的可能性。样本点  $s_i$  的完整性分数  $g_c$  计算为

$$g_c(s_i) = w_i \cdot \nabla(s_i) \cdot \mathbf{n}_i \quad (1)$$

根据泊松曲面重建算法，如果曲面质量高并且与初始点云一致，则等式 (1) 将具有更高的值。相反，如果局部点云中存在孔洞，则结果将导致较低的分值。

平滑度分数。平滑度分数表示初始点云在样本点周围的局部法线变化，对于样本点  $s_i$ ，平滑度得分  $g_s(s_i)$  定义为

$$g_s(s_i) = \sum_{p_j \in \mathcal{N}_i} w_{p_j} \cdot \exp \left\{ \frac{-\left(1 - \vec{N}_{p_j} \cdot \mathbf{n}_i\right)^2}{(1 - \cos \theta_t)^2} \right\} \quad (2)$$

$$w_{p_j} = \exp \left\{ -\frac{\|s_i - p_j\|_2}{R} \right\}, \quad (3)$$

其中  $R$  是泊松盘采样中定义的样本半径； $w_{pj}$  为距离权重，与初始点云中样本点  $s_i$  与点  $p_j$  的欧氏距离有关； $\theta_t$  是预定义的角度阈值。平滑度分数越高，表示初始点云中该样本点周围的局部区域越平滑；相反，分数较低表明局部表面重建质量相对较差，可能包含更多需要仔细重新扫描的几何细节。样本点  $s_i$  的最终质量分数是使用这两个指标的乘积计算得出的：

$$g(s_i) = g_c(s_i) \cdot g_s(s_i) \quad (4)$$

最后对所有样本点的质量得分进行归一化：

$$g^*(s_i) = \frac{g(s_i) - \min_{k \in S} \{g(s_k)\}}{\max_{k \in S} \{g(s_k)\} - \min_{k \in S} \{g(s_k)\}}. \quad (5)$$

### 3.3 Aerial Viewpoint Generation

#### 3.3.1 View Sampling Space

给定最近距离  $d_{min}$  和最远距离  $d_{max}$ ，即无人机到场景物体的安全距离，可以得到受限的视图采样空间来为无人机选择视点  $V$ 。当与物体的距离小于  $d_{min}$  时，也可以定义禁飞空间。如图 3 所示。

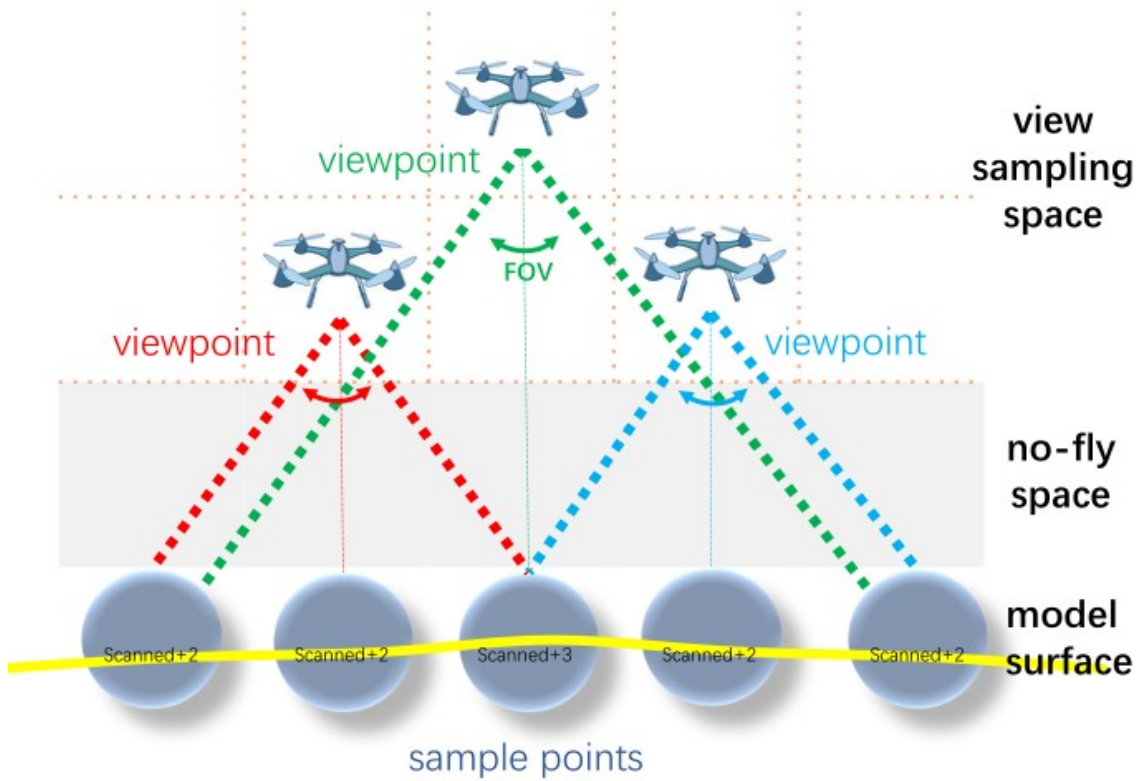


图 3: 视图采样空间

#### 3.3.2 Viewpoint Selection

为了降低计算的复杂度，作者将视图采样空间划分大小相等的体素，每个体素的中心作为一个候选视点。视点选择的目的是从所有的候选视点中选择一组视点  $V$ ，这些视点能够充分观察场景，为了实现这一目标，作者提出了两种候选视点：第一类是重新扫描不完整和低质量区域的局部视点  $v_l$ ；第二种是全局视点  $v_g$ ，对应整体覆盖。

局部视点。通过前面计算的初始点云的质量，可以确定质量分数较低的样本点是需要比较高的优先级进行扫描的，所以可以根据质量分数对所有的样本点进行排序，低于阈值的采样点集合为  $S_l$ ，对



于每个  $S_l$  中的采样点  $s_j$ ，找到最佳对应的候选视点。样本点  $s_j$  和候选视点  $v_i^c$  的置信度计算主要从三个方面进行度量：

(i) 采样点  $s_j$  必须是能被  $v_i^c$  可见的，即 UAV 是可以从视点能够直接观察到该样本点，需要注意的是，在实践中，默认视点上方的采样点无人机是无法直接观察得到的。作者定义了一个可见权重参数  $w_v(v_i^c, s_j)$ ，如果是可见的，那么  $w_v(v_i^c, s_j) = 1$ 。

(ii) 欧氏距离  $d_{ij} = \|v_i^c - s_j\|$  是代表了采样点到候选视点之间的距离，定义一个最佳距离  $d_{opt}$ ，它的值为预定义的安全距离的最小值  $d_{min}$  与最大值  $d_{max}$  的平均值  $d_{opt} = (d_{min} + d_{max}) / 2$ ，距离置信度  $c_d$  的计算公式如下：

$$c_d(v_i^c, s_j) = \begin{cases} 1 - \frac{(d_{ij} - d_{opt})^2}{(d_{opt} - d_{min})^2}, & d_{min} \leq d_{ij} \leq d_{max} \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

(iii) 视线与采样点法线之间的角度应该在阈值  $\theta_t$  内，方向的置信度计算如下：

$$c_o(v_i^c, s_j) = \exp \left\{ - (1 - \mathbf{n}_s \cdot \mathbf{d}(v_i^c, s_j))^2 / (1 - \cos \theta_t)^2 \right\}, \quad (7)$$

其中， $\mathbf{d}(v_i^c, s_j) = \frac{v_i - s_j}{\|v_i - s_j\|}$  表示从样本点  $s_j$  到候选视点  $v_i^c$  的向量， $\mathbf{n}_s$  表示样本点该处的法向量。

最后对于每一对视点-采样点对，我们可以获得一个置信度分数：

$$c(v_i^c, s_j) = w_v(v_i^c, s_j) \cdot c_d(v_i^c, s_j) \cdot c_o(v_i^c, s_j). \quad (8)$$

使用贪心法迭代选择具有高置信度分数的未选择候选视点。

全局视点。当确定了局部视点集合之后，将采样点距离邻居采样点距离小于  $|\mathbf{d}(v_i^l, s_j)| \cdot \sin(\theta_{fov} / 2.0)$  标记为 1。所有的样本点都至少被扫描两次，对于那些标记小于 2 的采样点，使用与前文提到的选择方法，从剩下的候选视点中进行选择全局视点。

### 3.4 Path Planning

经过前面的步骤，得到视点的合集，对这些视点集合设计一条快速并且平稳的路径，作者提出了一个视点与视点之间的成本函数。

(1) 距离成本  $c_t(v_i, v_j)$ ：与两个视点  $v_i$  和  $v_j$  之间的距离有关，计算公式如下：

$$c_t(v_i, v_j) = \frac{b(v_i, v_j) - b_{min}}{b_{max} - b_{min}} \quad (9)$$

其中， $b(v_i, v_j)$  是两个视点之间的距离， $b_{min}$  是所有视点与视点之间的最短距离， $b_{max}$  是所有视点与视点之间的最大距离。

(2) 旋转成本  $c_r(v_i, v_j)$ ：与视点之间方向的角度有关，可以计算为：

$$c_r(v_i, v_j) = \frac{1}{2} \left( 1 - \frac{\mathbf{o}_i \cdot \mathbf{o}_j}{\|\mathbf{o}_i\| \|\mathbf{o}_j\|} \right) \quad (10)$$

视点方向之间的最小角度定义为  $0^\circ$ ，而最大角度为  $180^\circ$ 。角度越接近  $0^\circ$ ，旋转代价越接近 0；当角度越接近  $180^\circ$  时，旋转成本越接近 1。

分别计算以上两种成本之后，最终转换成本可以得到如下式子：

$$c(v_i, v_j) = 0.5 \cdot c_t(v_i, v_j) + 0.5 \cdot c_r(v_i, v_j) \quad (11)$$

为了生成最小成本的连续路径，作者将问题表述为 TSP 问题，视点集合中的视点被当作城市，使

用蚁群优化算法 (ASO) 进行优化。

## 4 复现细节

### 4.1 与已有开源代码对比

网上目前还没有关于这篇论文公开的源代码。在实现的过程中，使用到了英特尔开发的 `embree` 库来实现判断光线是否与模型面发生碰撞，代码层面使用到了一些官方公布的 `demo`。在实现的过程中，需要注意的地方还是挺多的。设计采样点和视点的数据结构，采样点数据结构包含位置、法线方向、质量和标记四个属性，视点数据结构包含位置、法线方向、采样点法线的角度和分数四个属性。设计图的数据结构，主要包含了模型的三维空间中 3 个方向上的最小值和最大值 (即：模型的边界)，x 和 y 方向的偏移量，x 和 y 方向上的分辨率和一个维护高度的二维向量 `heighmap`。

现有的基准模型中包含了粗糙模型了，所以可以直接在这个粗糙模型表面上采样，使用 `meshlab` 进行泊松采样，导出带有坐标、法线和质量的采样点 `ply` 文件。在代码实现的过程中，将提前处理好的 `ply` 文件进行读入，同时，为了获取模型面的信息以进行后续的可见性判断，也应该要把原始模型的 `ply` 文件读入。

接着初始化所需要用到的信息，包括图信息数据结构，工具包等。初始化完成之后，在可行空间内针对前面得到的每个采样点周围初始化一些视点，并且保证这些视点不在模型的内部，可以对视点坐标与对应的一开始前面得到的 `heighmap` 值进行对比，如果是小于，就会对其进行修正，采样点的位置和法线方向不一样，生成初始视点的位置和方向也会有所不同的，每个采样点的初始化视点都会放入到一个集合中。

当所有的采样点都进行有初始化视点之后，就开始对每个采样点的每个视点进行筛选。进行筛选前，会对采样点的所有初始视点进行可见性判断，使用前面所提到英特尔开发的 `embree` 库，对那些不能正常见到采样点的视点，会进行排除。将进行可见性判断之后得到的视点，用前面所提及的置信度分数公式进行逐一的计算并进行由大到小的排序。对于每个采样点，都应该至少保证有两个视点能够正常地观察到该采样点以确保能够进行完整的重建。

前面提及到可见性判断过程中，除了判断每个视点能够沿着射线直接看见的采样点之外，也需要考虑由于无人机搭载的是相机进行拍摄，相机可以拍摄一定范围内其它的采样点。因此，对于同一个视点，可以让其它采样点从世界坐标系中采样点的空间坐标转换到该视点位置和方向的相机坐标系下的相机坐标，最后转换成图像区域的像素坐标，判断该像素坐标是否在合理的范围内，如果在，说明该视点能够观察到该采样点。

### 4.2 实验环境搭建

这次复现使用的编程语言是 C++，操作系统是 win10，C++ 版本为 C++11，跨平台开源软件包管理器 `vcpkg`，代码编辑器 `CLion`。

需要安装的 C++ 库包括几何计算库 `CGAL`、线性代数计算库 `Eigen3`、光线追踪库 `embree3`、点云文件处理库 `tinyply`。以上库的安装都需要安装 windows 64 位操作系统的版本。

可以根据网上的教程在 `github` 上拉取 `vcpkg` 的源码进行安装运行，然后打开命令行窗口，输入 `vcpkg install` 库名进行安装和编译相关的库，注意要装适配自己本机的版本。在 `CLion` 项目设置 `CMake`

为 vcpkg 的 cmake，同时配置好 CMakeList.txt 文件。

## 5 实验结果分析

根据代码实际运行生成的视点，在拍照程序中对场景进行拍照，验证生成的视点是否是合理并且有效的，这里使用了两个场景，一个是 NY-1 场景和 School 场景。部分场景拍照的效果如图所示。

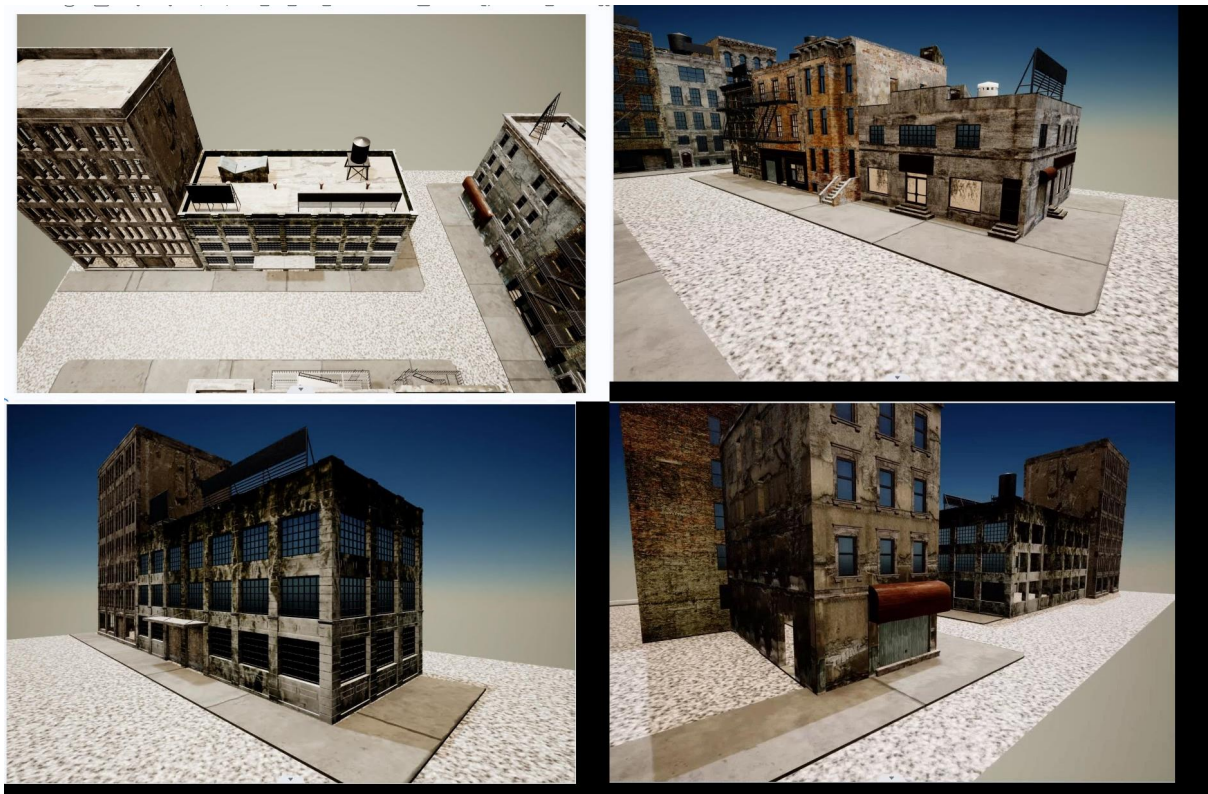


图 4: NY-1 场景部分拍照图

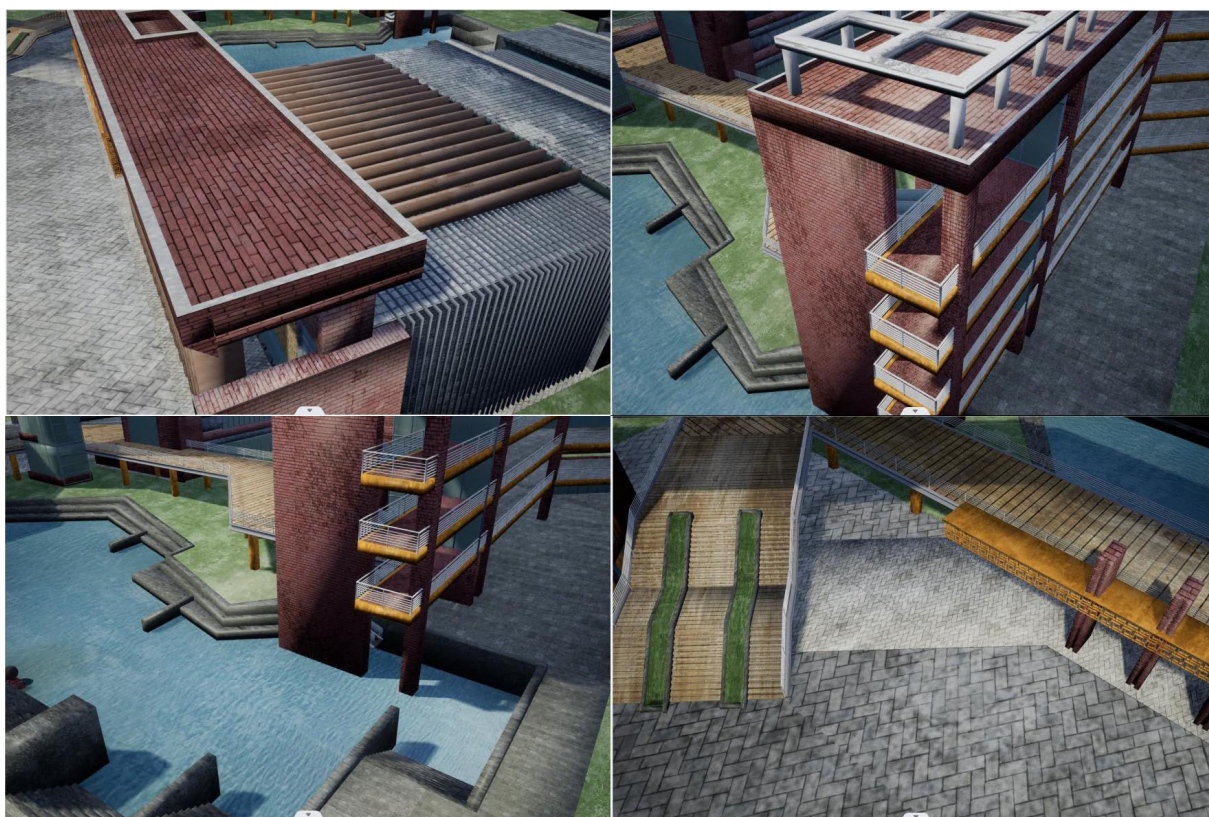


图 5: School 场景部分拍照图

对比其它的实验结果，这次复现的效果没有能比作者做的更好，比如总的拍照数量比作者多，作



者大概使用了 250 副图左右能实现比较好的对 NY-1 场景的拍摄和重建，而本次复现的方法生成视点数量接近 800，远多于作者的，可能的原因是一些视点的冗余度较高没有很好的剔除掉，另一方面是有些视点生成的位置不是很合理(但是，在 School 场景中这种情况几乎没有)，通过这些视点所拍摄的照片重建的精确度和完整性没有达到比较好的效果。

## 6 总结与展望

通过预先得到的粗糙模型，在粗糙模型上面进行采样，评估初始点云的质量。判断视点与采样点之间的可见性，计算视点与采样点之间的置信度分数，根据视点与采样点之间的置信度从大到小进行选择合适 viewpoint，最终形成一个合适的 viewpoint 集。

自己在实现的过程中阅读了这个方向相关的一些资料和论文，也学习到了一些视点生成和选择的方法。由于现阶段网上关于这方面基本上没有什么公开的源代码，所以自己在实现的时候也碰到挺多困难的。在实现过程中，自己 viewpoint 选择的实现做的不够好，有些视点的冗余度较高但是没有去除掉导致自己最后生成的 viewpoint 比原作者的多了很多，可见性判断那部分，可能是由于一些参数没调好的原因或者光线追踪部分代码写的有些问题，导致自己的代码在生成的一些 viewpoint 在场景模型 (NY-1 场景) 的内部，这也是不合理的，还有就是实现的算法的时间复杂度较高，采样点如果较多的话，也会让后面 viewpoint 的生成和选择时间会比较久，导致整个代码运行的比较久。

未来会继续进一步研究无人机 viewpoint 的选择这一方向，如何在具有较多的约束条件下，在安全空间内，尽可能的去寻找一个尽可能小的 viewpoint 集合来让无人机能够在这些位置和角度拍摄照片并且能够覆盖到整个场景，使三维重建的质量达到一个好的效果。

## 参考文献

- [1] FERNÁNDEZ-HERNANDEZ J, GONZÁLEZ-AGUILERA D, RODRÍGUEZ-GONZÁLVEZ P, et al. Image-Based Modelling from Unmanned Aerial Vehicle (UAV) Photogrammetry: An Effective, Low-Cost Tool for Archaeological Applications[J]. Archaeometry, 2015, 57(1): 128-145.
- [2] MANCINI F, DUBBINI M, GATTELLI M, et al. Using Unmanned Aerial Vehicles (UAV) for High-Resolution Reconstruction of Topography: The Structure from Motion Approach on Coastal Environments[J]. Remote Sens, 2013, 5(1): 6880-6898.
- [3] CADENA C, CARLONE L, CARRILLO H, et al. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age[J]. IEEE Trans. Robot, 2016, 32(1): 1309-1332.
- [4] SUZUKI T, AMANO Y., HASHIZUME T, et al. 3D Terrain Reconstruction by Small Unmanned Aerial Vehicle Using SIFT-Based Monocular SLAM[J]. J. Robot. Mechatron, 2011, 23(1): 292-301.
- [5] TEIXEIRA L, CHLI M. Real-time local 3D reconstruction for aerial inspection using superpixel expansion[C]//In Proceedings of the 2017 IEEE International Conference on Robotics and Automation. 2017: 4560-4567.
- [6] KRIEGEL S, RINK C, BODENMÜLLER T, et al. Next-Best-Scan Planning for Autonomous 3D Modeling.[C]//In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and



Systems. 2012: 7-12.

- [7] WU S, SUN W, LONG P, et al. Quality-Driven Poisson-Guided Autoscanning[J]. ACM Trans. Graph, 2014, 33(1).
- [8] XU K, HUANG H, SHI Y, et al. Autoscanning for Coupled Scene Reconstruction and Proactive Object Analysis[J]. ACM Trans. Graph., 2015, 34(6): 177:1-177:14.
- [9] ROBERTS M, SHAH D, S and Dey, NARR A, et al. Submodular Trajectory Optimization for Aerial 3D Scanning.[C]//In Proceedings of the IEEE International Conference on Computer Vision. 2017: 5334-5343.
- [10] PENG C, ISLER V ., DEY D, et al. Adaptive View Planning for Aerial 3D Reconstruction.[C]//In Proceedings of the 2019 International Conference on Robotics and Automation. 2019: 2981-2987.
- [11] HEPP B, NIESSNER M, HILLIGES O. Plan3D: Viewpoint and Trajectory Optimization for Aerial Multi-View Stereo Reconstruction[J]. ACM Trans. Graph., 2018, 38(1): 1-7.
- [12] SMITH N, MOEHRLE N, GOESELE M, et al. Aerial Path Planning for Urban Scene Reconstruction: A Continuous Optimization Method and Benchmark[J]. ACM Trans. Graph., 2018, 37(6).
- [13] QU Y, HUANG J, ZHANG X. Rapid 3D Reconstruction for Image Sequence Acquired from UAV Camera[J]. Sensors, 2018, 18(1). DOI: 10.3390/s18010225.
- [14] RUBLEE E, RABAUD V, KONOLIGE K, et al. ORB: An efficient alternative to SIFT or SURF[J]., 2011: 2564-2571. DOI: 10.1109/ICCV.2011.6126544.
- [15] MUR-ARTAL R, TARDÓS J D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras[J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-1262.