arXiv:2304.03763v2 [cs.CV] 1 Sep 2023

使用视图一致性修复进行3D场景中的杂物检测与移除

Fangyin Wei Thomas Funkhouser SzymonRusinkiewicz 普林斯顿大学

摘要

从场景中移除杂物对于许多应用至关重要，应用范围从关注隐私的内容过滤到数据增强。在本文中，我们提出了一种自动系统，可从*3D*场景中移除杂物，并使用一致的几何与纹理进行图像修复。我们为该系统的两个关键组件提出了相应技术：基于共享属性的三维分割和三维修复，这两者都是重要的问题。计算机视觉中常见的物体类别无法很好地刻画三维场景杂物（频繁移动物体）的定义。为解决缺乏定义明确的杂物标注这一问题，我们将嘈杂的细粒度标签进行分组，利用虚拟渲染，并施加实例级面积敏感损失。移除杂物后，我们通过融合修复后的 *RGB-D*图像，修复所得孔洞中的几何与纹理。对于网格重建，这需要新颖的投票与剪枝策略，以保证各个单独修复图像之间的多视图一致性。在*ScanNet*和*Matterport3D*数据集上的实验表明，我们的方法在杂物分割和三维修复方面，无论视觉效果还是定量指标，均优于基线方法。项目页面：*https://weify627.github.io/clutter/*。

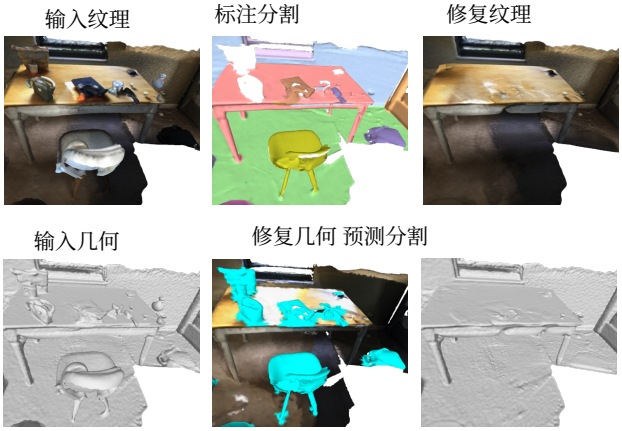

图 1. **利用视图一致性修复进行三维场景中的杂物检测与移除。**从输入场景（左侧）开始，我们的首要任务是预测杂物区域。传统的语义分割数据集和方法 [9]（上方中央）并非为解决这一问题而设计，且包含缺失或错误的标签（尤其是针对小物体），以及有时才与杂物对应的语义类别。相比之下，我们的方法（下方中央）能够正确识别杂物分割区域，随后将其移除，并使用连贯的纹理和几何进行修复（右侧）。

1. 引言

随着RGB-D相机的普及，我们可以想象这样一天：人们能够用手机快速扫描房间，并将重建的3D模型上传到网站上进行转租。展示一个没有枕头和厨房用具散落各处的整洁、吸引人的房间，会更令人满意。然而，人们通常不愿意在扫描前花费数小时清理这些物品。这时，用于自动移除场景杂物的工具就派上用场了。这种场景编辑工具可以应用于广泛的任务，从自动内容和隐私过滤，到数据增强（通过向清理后的场景中添加新物体）。

本文研究如何从一组室内RGB-D扫描中重建不含杂物的场景。该任务包括两个步骤：杂物分割和三维修复，而这两项任务都带来了新的挑战。

得益于大规模3D室内场景数据集 [4, 9,1, 41], ，数据驱动方法在3D场景分割方面取得了显著成效 [18, 35, 24]。然而，现有标注在杂物分割方面存在不足，原因有二。首先，即使是迄今为止粒度最细的数据集（基准数据集包含约550个原始类别 [4, 9], 200）[41]，也存在歧义。例如，类别为白板的一些物体可以移动，属于杂物；另一些则固定在墙上，不属于杂物。这种类别内部的歧义十分普遍：例如灯具、座椅、扬声器、装饰物、等等，它们既有固定（非杂物）形式，也有可移动（杂物）形式。其次，标注质量往往不理想，一些杂物未被标注，或被误标为非杂物，如图1上方中央所示。考虑到网格分辨率以及标注大量小物体所需的时间，要在大型数据集中精确、全面地标注杂物几乎是不可能的。

三维修复任务旨在合成视觉上

并对三维场景缺失区域进行几何上合理的补全。尽管一些已有工作 [42, 47, 32]会对不完整扫描或遮挡造成的较小孔洞进行图像修复，但我们关注的是物体移除所产生的较大孔洞。复杂的几何和纹理使这类孔洞的图像修复更具挑战性。一种简单的解决方案是在剩余点上运行三维孔洞填充或网格重建算法。然而，这些算法只考虑局部几何和低级纹理信息，而在许多情况下，合理的补全需要语义理解。例如，移除覆盖桌角的一些杂物后，图像修复后的场景应恢复桌角，而不只是填补孔洞，却仍缺失桌角。直接处理三维场景需要同时应对相互交织的语义和复杂几何，这可能导致输出分辨率较低且包含许多伪影 [12, 21]。我们提出改为对 RGB-D 序列进行图像修复，然后重建最终网格。这样便将问题拆分为语义上合理的图像修复和几何填充。前者可利用现有先进的二维图像修复方法更轻松地解决，而且有望比仅在有限二维数据上训练的方法更强大。

在本工作中，我们致力于解决上述挑战，并提出一种用于场景杂物移除和三维修复的自动化系统。为了更有效地分割小型杂物，我们设计了面积敏感损失，促使模型关注通常只能获得有限损失信号的小型杂物。我们采用一种深度架构，同时将 RGB 图像和三维点作为输入，并将渲染的虚拟视图作为二维输入，以获得更大的表面覆盖范围。不同于以往在三维空间中识别物体后直接预测三维修复结果的方法，我们将物体掩码投影到 RGB-D图像上，并执行图像修复和图像引导的深度补全。为了强制不同帧之间保持一致性，从而获得质量更高的三维重建结果，我们在各帧之间执行新颖的一致性投票和剪枝。我们循环执行图像修复，直至所有缺失区域都被填充。最终得到的一组具有一致修复结果的 RGB-D图像可以合并为一个干净的最终场景。

我们在 ScanNet [9]和 Matterport3D [4]数据集上进行了大量消融实验，以验证系统设计。我们还将方法与三维分割和图像修复基线方法进行比较，展示了所提出模型的有效性。总而言之，我们的贡献包括：

- 一个自动化系统，可解决场景杂物移除与修复这一新任务。
- 一种采用面积敏感损失的三维分割方法，即使数据带有噪声，也能更好地分割小物体。
- 一种采用迭代式视图一致 RGB-D 三维修复的三维修复方法，性能优于先前方法。

2. 相关工作

3D语义分割 许多以往的工作研究了如何将三维点云分割为语义类别。这项工作专注于识别特定类型的物体类别 [9, 4, 1], , 因此通常在标签集较小 [18, 35, 24], 时表现良好，但当存在许多可能的类别，而训练集 [41]中的大多数类别只有少量样本可用时，表现就不那么理想。在本文中，我们研究了更为抽象的任务：识别杂物 与。非杂物，这一属性并不直接对应于物体类别标签。

RGB图像修复传统的单视图方法 [14, 13, 2, 3] （如基于图块或基于扩散的方法）通常能够取得逼真的效果，但缺乏高层次的图像理解能力。自开创性工作 [37]将对抗训练和编码器—解码器结构引入这一任务以来，人们提出了各种技术，包括多阶段生成 [34, 40], 、门控卷积 [52], 、循环方法 [26, 53], 等。较新的研究开始关注更具挑战性的设置，例如多样化图像修复 [55, 46] 和大孔洞填充 [54, 27, 44]。具体而言，LaMa [44]采用快速傅里叶卷积，并使用极大的掩码进行训练。该方法以相对较低的训练和推理成本实现了高保真效果。单图像三维摄影旨在利用基于图像的渲染技术 [20, 43, 23]合成新视图。从某种意义上说，它可以被视为从单张图像中填补其他视图里的小孔洞。

深度、多视图和网格修复近年来，基于深度学习的深度补全方法在有监督 [36, 17, 5, 51] 和无监督 [30, 49, 29] 学习方面都取得了巨大成功。特别是，NLSPN [36] 通过可变形卷积 [56]进行非局部深度传播。多视图图像修复通常被用作基于图像的渲染 [45, 25, 50, 39, 33]的一种技术。这些方法与我们的方法都涉及具有三维感知能力的图像修复。然而，我们旨在获得修复后的网格，而多视图图像修复旨在渲染新的视图，即，跨帧一致性可以作为一个目标进行优化，但无法得到保证。“修复”一词也被用于形状补全 [16, 42, 47, 32]。例如，基于示例的方法 [42] 旨在填补不完整扫描中的小孔洞。这些方法只关注几何。表面重建算法 [38, 22]只能在距离足够近的点之间创建表面，因此无法生成新的表面来填补大孔洞。

三维场景修复 三维场景修复旨在为三维场景中缺失的部分填充合理的几何结构（和纹理）。早期研究仅补全几何结构 [11, 10],, 而近期研究开始对几何结构和纹理进行三维修复

已受到更多关注 [12, 21, 31, 48]。基于 TSDF 的方法 [12, 21] 通过一次前向传播预测场景补全效果。它们擅长恢复主要结构（例如平坦墙面和地板），但往往难以处理更具挑战性的结构。相比之下，我们的基于网格的方法通过分离几何结构和纹理，迭代地细化场景。与我们的工作同期，有两种基于 NeRF 的方法 [31, 48] 也采用迭代细化来移除场景中的物体。一个关键区别在于如何处理多视图不一致性。我们设计了剪枝和投票技术来选择一致视图；Weder等人 [48]学习逐图像置信度，以重新加权 NeRF 损失；SPIn-NeRF [31] 既不选择视图，也不强制一致性。另一个区别在于如何修复深度。虽然 [31, 48]他们只根据被遮罩的深度修复深度，我们则采用 RGB 引导的深度修复，使修复后的深度与修复后的 RGB 图像更好地对齐。

3. 方法

给定一个 RGB-D 序列及其重建得到的室内场景网格作为输入，我们的方法输出一个不含杂物的三维场景（如图2所示）。第一步是预测 3D 杂乱物体分割（第3.1节）。然后，我们将每个杂物分割区域投影到各个相机视图中，以遮蔽颜色图像和深度图像中的杂物区域。接着进行彩色图像修复，其结果用于指导深度补全（第3.2.1节）。完成所有 RGB-D 帧的图像修复后，我们执行一致性剪枝和投票。如果移除杂物后仍有缺失区域，则将经一致性检查的修复结果送回图像修复和深度补全流程，以填补剩余孔洞（第3.2.2节）。这一循环会持续进行，直到填补所有缺失区域。最后，我们对一致性得到保证的修复结果执行泊松表面重建 [22]，以获得最终修复后的场景网格（第3.2.3节）。

3.1. 杂物分割

我们的重点是提出一种基于共享属性（例如，杂物之间的共享属性）而非常见基准语义类别的新型场景分割方法。因此，尽管杂乱定义的具体形式可以灵活调整（详见补充材料），但要检验我们的方法，明确的杂乱定义至关重要。定义杂物后，我们将所有原始类别（ScanNet 中的 549 类 [9]）划分为杂物、非杂物和混合类别。为减少歧义，我们排除混合类别，仅将其他类别用于粗粒度监督。需要注意的是，由于杂物具有异质性，大多数杂物类别各自包含的实例少于 20 个，且杂物的实例总数少于非杂物。

3.1.1 面积敏感损失

三维分割模型通常利用交叉熵损失监督三维预测。然而，直接应用标准交叉熵损失，无法训练出在我们的任务上达到与 [18]相近性能的模型。与一般物体相比，杂物的一个独特之处在于，尺寸较小的杂物占比更高。然而，标准的逐顶点损失过于关注较大的实例。为了让模型重新关注小物体，我们将标准交叉熵损失修改为以下形式：

$$L^*_{CE(3d)} = \frac{1}{N_v}\sum_{c\in C}\sum_{i\in O_c}\Big(w_i \sum_{j\in V_i} y_{(c,j)} log(p'_{(c,j)})\Big),$$

其中， N_v 是场景中的顶点总数， C是所有类别（杂物和非杂物）的集合， O_c 是当前场景中类别 c 的实例集合， V_i 是物体 i的顶点集合， $y_{(c,j)}$ 和 $p'_{(c,j)}$ 分别是顶点 j的真实标签和预测类别概率。特别地， w_i 是我们引入的用于平衡实例大小的权重，其定义如下：

$$w_i = \Big(\frac{median(N)}{n_i}\Big)^k, n_i \in N,$$

其中， N 是场景中所有物体的顶点数量列表， n_i 是物体 i的顶点数量， $k \geq 0$ 是用于控制权重作用强度的调制因子。较小的 n_i （即物体实例 i的顶点较少）会增大权重 w_i 以及物体 i对应的损失。较大的 k会增大 w_i ，从而加重对较小物体实例的损失惩罚。特别地，当 $k = 0$ 且 $w_i = 1$时， $L^*_{CE(3d)}$ 与交叉熵损失形式相同。

值得指出的是，我们的面积敏感损失表面上可能与中位频率损失 [15] 或平衡交叉熵 [19, 8]相似。然而，我们从根本上有所不同。其他损失旨在解决类别不平衡问题，而我们的损失作用于实例层面，以更好地分割小物体——无论该类别的总表面积如何，表面积较小的物体实例都会被赋予更高的权重。特别是，并非所有杂物实例都很小（其中有一些容易识别且尺寸较大的类别），并且也并非所有非杂物实例都很大。我们预计，我们的损失将有助于跨类别处理小物体。在我们的二元分割任务中，类别平衡损失只会简单地将某一类别统一赋予更大的权重。我们通过实验证实，这并不能改善我们任务的结果。

3.1.2 实现与虚拟视图渲染

我们的重点是为异构小物体设计有效的损失函数。我们选择BPNet [18]作为骨干三维分割模型。它包含通过双向特征投影连接的二维和三维子网络，该投影采用

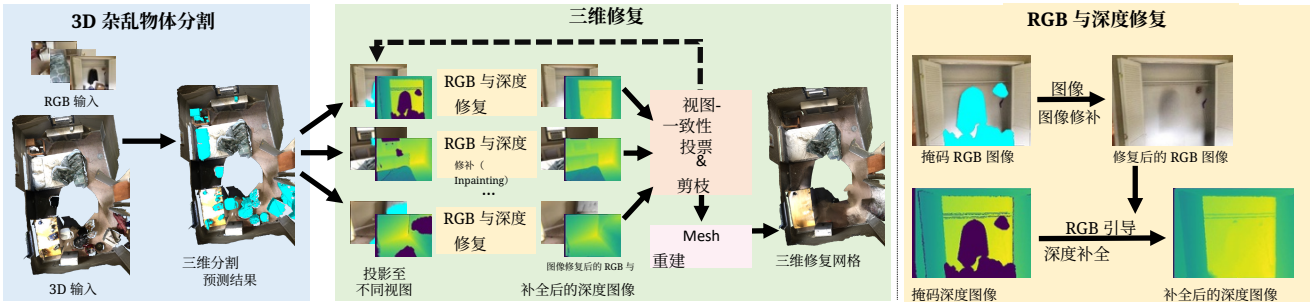

图 2. **方法概述**。 给定 RGB-D 重建结果和 RGB 帧，我们首先预测 3D 杂乱物体分割。随后，将预测结果投影到不同的相机视角中，以遮罩 RGB 和深度图像。接下来，我们对 RGB 图像进行图像修复，并利用修复结果引导深度补全。之后，我们对所有 RGB-D 帧执行一致性剪枝和投票。仍有缺失区域的任何图像修复结果都会被送回，再次进行 RGB 和深度图像修复。循环会持续进行，直到所有缺失区域都被填充。最后，我们通过网格重建获得最终修复后的场景网格。

分别作为输入的彩色图像和点云。损失函数 $L_{seg} = L^*_{CE(3d)} + \lambda L^*_{CE(2d)}$ 是对三维和二维预测的面积敏感损失进行加权求和，并由权重 λ进行平衡。我们遵循 [18]，每次输入 3 个视图。然而，最初捕获的三张图像仅覆盖整个场景的一小部分表面，而增加输入视图的数量会大量消耗内存。受 [24], 启发，我们用虚拟渲染结果替代最初捕获的图像，并可通过设置相机参数来增加观测上下文。

3.2. 三维修复

三维修复是一项新近兴起的任务 [12, 21], ，而三维修复中的一个关键挑战是几何与语义（或外观）之间复杂的相互作用。为解决这一问题，我们的核心思想是利用中间二维表示，将带纹理的三维场景解耦为RGB 图像（外观）和几何信息，从而可以更容易地对其进行图像修复，随后用于多视图三维重建。值得注意的是，这种针对三维修复问题的新颖二维方法不同于以往依赖三维卷积直接在三维空间中进行修复的方法 [12, 21]。下文中，我们将首先简要概述所提出的流程，然后介绍用于应对这些挑战的新技术。

3.2.1 单帧修复

我们首先将来自三维网格的杂物预测投影到每个视图上，为 RGB 图像和深度图像创建二维掩码。我们提出的二维方法的一个优势是能够利用强大的二维合成模型。这也有可能提升泛化能力，尤其是在面对有限的三维数据（相较于图像数据）时。

彩色图像修复 我们使用 LaMa [44]及其公开发布的预训练权重修复每张彩色图像。该模型在多样化的数据上训练，因此能够很好地泛化到室内拍摄的图像。不过，当存在多种合理的补全结果时（尤其是掩码较大时），模型往往会从不同视角为同一区域填入不同内容。这也会导致深度补全不一致，相关问题将在第3.2.2节讨论。

深度补全 现有的深度补全方法 [36, 17, 5, 51, 30, 49, 29] 尝试根据均匀且稀疏采样的像素预测完整的深度图，例如，使用局部传播。然而，我们需要补全带有（可能很大的）孔洞且缺乏局部信息的稠密图。我们采用 NLSPN [36] 来进行非局部深度传播，以实现图像引导的深度补全。由于我们的任务没有标注数据，我们设计了一种方法，在原本干净的深度图上创建逼真的孔洞，以生成监督信号：

我们假设 m_1 是捕获的深度图 d_1 的粗略杂物掩码。我们使用第二个视角中的杂物掩码 m_2 来遮盖 d_1 上的区域。同时，我们确保 m_1 中原本被标记为杂物的区域不会被遮盖。通过从另一个视角复制并粘贴 m_2，我们确保训练期间使用的掩码具有杂物的真实边界。通过不遮盖 m_1 中的杂物，我们保证深度补全模型无需从空洞区域中臆测出杂物。

3.2.2 受视图一致性启发的细化

图像修复和深度补全的巨大进展，使得对单个帧进行高质量图像修复成为可能。然而，由于图像修复的多样性以及每个模型的精度限制，不同帧之间会出现不一致（图6）。这种不一致会在网格重建中造成许多噪声。在本节中，我们将讨论该任务的独特特征以及

解释我们如何基于观测结果强制保持一致性。

单帧一致性剪枝 物体移除任务本身决定了，同一区域内图像修复得到的深度 d_{pre} 应始终大于原始采集深度 d_{cap}。这里的区域包括每个像素 p 以及每个连通掩码区域 i。基于这一观察，我们有

$$d^1_{con}(p) = \begin{cases} d_{pre}(p) & \text{if } \ d_{cap}(p) < d_{pre}(p), \\ 0 & \text{else;} \end{cases}$$

$$d^2_{con}(i) = \begin{cases} d^1_{con}(i) & \text{if } \ d_{cap}(i) < d_{pre}(i), \\ 0 & \text{else,} \end{cases}$$

其中， d^1_{con} 和 d^2_{con} 分别是针对像素和连通区域进行一致性检查得到的深度输出。由于我们关注的是填补物体移除造成的孔洞，而不是移动传感器导致的采集深度图像中的裂缝，因此会屏蔽 d_{pre} 中在 d_{cap}里为空的所有像素。我们还会舍弃面积超过整幅图像一半的修复区域，因为仅凭有限输入像素得到的修复结果极不可靠。

跨帧一致性剪枝 与空间雕刻的思路相同，将源视图s的图像修复深度 $d^{(s)}_{pre}$ 变形到目标视图 t后，不能遮挡该视图中的采集深度 $d^{(t)}_{cap}$。这意味着对于每个像素 p:

$$d^{3(s)}_{con}(p) = \begin{cases} d^{2(s)}_{con}(p) & \text{if } \ d^{(t)}_{cap}(p) < \text{Warp}^{s-t}\left(d^{(s)}_{pre}(p)\right), \\ 0 & \text{else,} \end{cases}$$

其中， $d^{3(s)}_{con}$ 是通过跨帧一致性剪枝得到的视图 s 下的深度输出。$\text{Warp}^{s-t}(\cdot)$ 是从视图 s 到视图 t的变换。给定两个视图的相机内参和位姿，即可计算该变换。对于每个视图对（s，t）执行此操作，其中 s，$t \in \{0, \ldots, N_s - 1\}$， N_s 是序列长度。

跨帧一致性投票到目前为止，我们已根据与捕获的深度的一致性，剔除了一些区域。然而，修复区域之间仍然存在不一致。其原因包括：对于来自不同视角的同一区域，（1）彩色图像被图像修复为不同的内容，或（2）即使彩色图像修复结果相同，补全后的深度也不同。为解决这一问题，我们使用预测的深度，将所有其他视角中的修复区域变形到视角 s。然后，我们考虑所有落入像素 p的变形后的深度值。对于视角 s，像素 p 处的深度值仅在距离以内的投影深度数量阈值 α from $d^{(s)}_{pre}(p)$ 超过落入 at p 的投影深度总数的 β %。因此我们有

$$d^{4(s)}_{con}(p) = \begin{cases} d^{3(s)}_{con}(p) & \text{if } \ r_\alpha > \beta\%, \\ 0 & \text{else,} \end{cases}$$

$$r_\alpha = \frac{\sum_t \mathbb{I}\left(|d^{(s)}_{pre}(p) - \text{Warp}^{t-s}\left(d^{(t)}_{pre}(p)\right)| < \alpha\right)}{\sum_t \mathbb{I}\left(\text{Warp}^{t-s}\left(d^{(t)}_{pre}(p)\right) > 0\right)},$$

其中， $t \in \{0, \ldots, N_s - 1\} \setminus \{s\}$、 $d^{4(s)}_{con}$是视角 s 处通过跨帧一致性投票得到的深度输出， $\mathbb{I}(\cdot)$是一个二值指示函数，其中 $\mathbb{I}(x) = 1$ 在 x 为真时取值，否则为0。

迭代细化与效率 这些一致性检查会在 $d^{4(s)}_{con}$中产生新的孔洞。我们使用这些新孔洞对当前步骤中的图像修复彩色图像和深度图进行遮罩，并使用新的遮罩 RGB-D图像重新开始三维修复。单帧图像修复和一致性细化会迭代执行，直至所有孔洞都被填充。跨帧剪枝或投票的复杂度为 $O(N_s^2)$。我们在Py-Torch 中实现了支持图形处理器的批量变形函数，该函数在NVIDIA TITAN 图形处理器上处理一个包含 200 帧的序列约需 7 分钟。

3.2.3 网格重建

为获得最终的图像修复三维场景，我们对其运行泊松表面重建 [22]，使用第 3.2.2 节中的修复结果。我们最多使用 1600 万个点，以及修复结果中小于 3.5 米的深度值。表面重建的树的最大深度设为 10。

4. 实验

4.1. 实验设置

数据集 我们使用 ScanNet 数据集 [9] 的官方划分。由于由分组原始类别生成的杂乱标签噪声很大，原始测试集无法提供准确的评估。我们使用 [9]中的标注工具，为 8 个具有不同设置的代表性测试场景手动标注。对这个干净标注集的评估能够真实反映模型性能。更多详情请参阅补充材料。

实现细节 我们将分割体素尺寸设为 5 厘米，损失平衡权重 λ 和调制因子 k 分别根据经验设为 0.3 和 1。

我们将 $\alpha = 0.05$， $\beta\% = 30\%$ 设置为用于跨帧一致性投票。

表 1. **杂物分割方法比较**。我们评估交并比，并将我们的方法与 BPNet 基线进行比较。

方法	原始测试集（含噪声）			手动测试集（干净）		
	交并比(NC)	交并比(C)	mIoU	交并比(NC)	交并比(C)	mIoU
BPNet [18]	0.87	0.43	0.65	**0.85**	0.35	0.60
Ours	**0.90**	**0.59**	**0.75**	0.84	**0.58**	**0.71**

表 2. **三维修复方法的比较**。我们使用二维和3D指标对基线方法进行定量比较。

方法	L1(↓)	L2(↓)	峰值信噪比(↑)	结构相似性(↑)	LPIPS(↓)	倒角距离(cm)(↓)
SPSG [6]	0.077	0.139	16.810	0.696	0.419	21.354
FF [22]	0.038	0.071	21.762	0.861	0.213	1.781
Ours	**0.018**	**0.033**	**23.673**	**0.918**	**0.151**	**0.621**

4.2. 基线方法与评估指标

杂物分割 我们将其与一个基线进行比较，该基线训练原始的多类别 BPNet [18]以分割 549 个原始类别，然后根据我们的杂物/非杂物分组对预测结果进行合并。其他所有设置均与我们的模型相同。我们使用虚拟视图，在两个测试集上评估两种方法的 3D 平均交并比。

三维修复 我们将其与 MeshLab 中的孔洞填充算法“Close Holes”进行比较 [7, 6] 以及泊松表面重建（PSR）[22]，设置与第 3.2.3节相同 [12]。我们进一步与两种近期的带纹理场景补全方法（SPSG [12] 和 FF [21]）进行比较。由于它们都是基于 Matterport3D 数据集开发的 [4]，我们还在 Matterport3D 上重新训练了我们的方法，并与它们的官方代码和模型权重进行比较。

4.3 结果

4.3.1 完整流程

在图 3中，我们展示了物体移除和三维修复完整流程的定性结果，并与三维修复基线方法进行了比较。中间三列展示了我们预测的 3D 杂乱物体分割、图像修复后的纹理和几何。右侧则将我们的结果与泊松表面重建（PSR）得到的几何 [22] 以及孔洞填充算法进行了比较。

第三列显示，我们的分割方法不仅能准确预测较大且形状规则的类别（如咖啡桌（最后一行）和椅子），也能很好地处理形状不规则且出现频率极低的物体。我们还发现，分割基线方法经常会漏掉小物体。由于训练集中的标注噪声较大，检测这些小物体并非易事（图 1）。我们之所以能成功检测出这些物体，主要得益于面积敏感损失；我们将在第 4.4.1 节中对此进行深入研究。

第四列展示了我们的方法能够用一致的纹理填充缺失区域。例如，在第四行中，桌子上杂物的深色在经过图像修复后全部消失了。我们将此归因于图像修复强大的语义理解能力。然而，仍然存在修复纹理伪影，其中相当一部分是由光照造成的。以第三行为例，我们能够成功分割出整盏灯具，但它投射出的灯光仍然存在，这并不理想。

第五列显示，我们的方法在闭合由物体移除造成的孔洞以及利用几何上合理的表面进行补全方面表现更好。例如，在第四行中，左侧的桌子有一大片缺失区域，这是由椅子的遮挡造成的。PSR 无法填补这个大孔洞，而孔洞填充算法错误地通过将桌面与桌腿连接起来填补了孔洞。相比之下，我们的方法不仅填补了整个孔洞，还恢复了桌面缺失的部分。在其他场景中也可以发现类似现象，例如第一行床上的枕头区域以及最后一行沙发上的枕头区域。

4.3.2 分割

我们在表1中报告了基线和我们的方法的交并比评估结果。结果表明，我们的方法优于所比较的 BPNet [18] 多类别分割模型。这并不太令人意外，因为所比较的模型最初应用于包含 20 个类别的任务 [18] 而我们的实验中有 549 个类别。正如先前的工作所指出的 [41]，大量类别会导致严重的类别不平衡。总之，该比较说明，杂物分割问题无法通过直接的细粒度分割模型解决。关于引入分割中的各个组件重要性的更多分析，请参见第4.4.1节。

4.3.3 三维修复

定性评估我们在图中将本方法与最新的带纹理三维场景修复方法进行比较。 4。两种方法都使用三维卷积，直接从不完整的输入体数据中预测 TSDF。它们受限于低分辨率，这是采用体表示方法的典型弱点。相比之下，我们的方法输出高分辨率网格。在第二列中，SPSG [12] 和 FF [21] 无法修复沙发上的孔洞，而我们的方法恢复出了与输入一致的颜色和几何。我们提出的二维方法的另一个优势是鲁棒性更强且泛化能力更好。例如，图 4的第一列展示了一个带有大孔洞的室外场景示例，这类场景在训练期间很少出现。然而，由于我们的二维修复模块接触过更多样化的训练示例，因此我们的方法也能很好地处理不常见的示例。

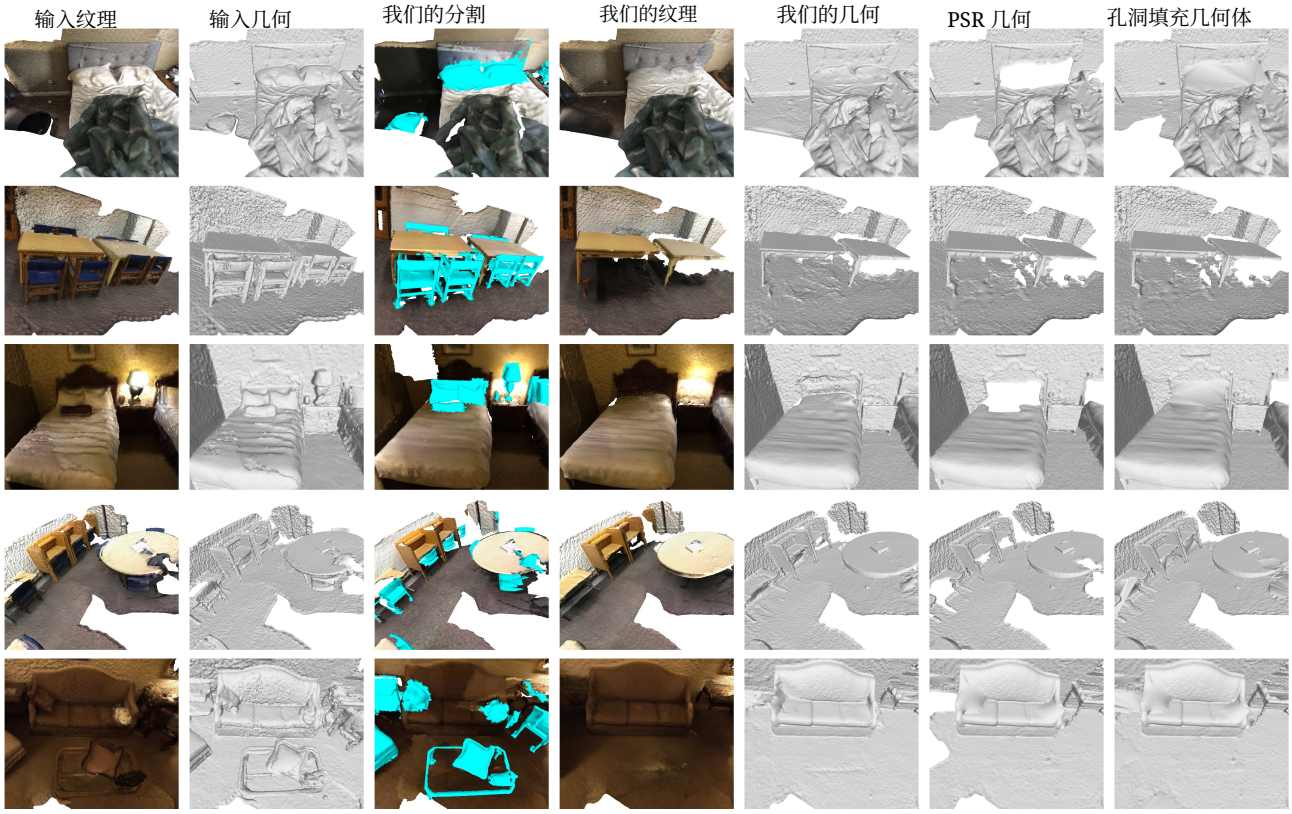

图 3. **自动杂物移除与三维修复的结果**。每行展示输入三维场景的一个渲染视图、我们的杂物分割结果，以及场景中颜色和几何均经过图像修复后的渲染视图。我们将后者与移除杂物后，分别使用泊松表面重建（PSR）或基于三角剖分的孔洞填充算法填补所产生孔洞的结果进行比较。

定量评估 为定量比较我们的方法与基线方法，我们创建了一个带有合成孔洞的数据集（详情见补充材料）。为了评估修复后的网格纹理，我们遵循图像修复中的常见做法，在修复后的网格的渲染结果上报告 L1、L2、峰值信噪比、结构相似性和 LPIPS [52, 55]。为了评估修复后的网格几何，我们比较原始表面与重建表面之间的 Chamfer 距离。我们在表2中报告评估结果。可以看到，我们提出的方法在所有指标上均优于基线方法。这表明我们的方法能够生成更好的纹理和几何，如图 3 和图 4所示。

4.4. 消融实验

4.4.1 杂物分割

在表3中，我们研究第3.1节介绍的各项策略所产生的影响。对于原始测试集和手动测试集，我们均报告非杂物（NC）类和杂物（C）类的交并比，以及杂物类的精确率和召回率。我们将重点关注手动测试集，因为其杂物标注更为准确。

表 3. **杂物分割的消融实验**。我们研究面积敏感损失在不同 k 设置下的效果。此外，我们还比较训练时使用渲染的 ($r.$)/ 采集的($c.$)二维视图，以及新的($n.$)/原始的($o.$)相机参数所带来的影响。(a-e) 在 $r./n.$ 个视图上进行评估，(f-h) 在 $c./o.$ 个视图上进行评估。(a-e) 中最佳、次佳和第三佳的结果分别以**粗体**、实线下划线和虚线下划线标出。

ID	k	2D 类型 /相机	原始测试集（含噪声） 交并比(NC)	交并比(C)	精确率	召回率	手动测试集（干净） 交并比(NC)	交并比(C)	精确率	召回率
(a)	0	$r./n.$	**0.94**	**0.67**	**0.85**	0.76	0.83	0.48	**0.91**	0.50
(b)	0.5	$r./n.$	0.93	0.65	0.79	0.79	**0.85**	0.55	0.88	0.59
(c)	1.0	$r./n.$	0.90	0.59	0.67	**0.83**	0.84	0.58	0.82	0.66
(d)	1.5	$r./n.$	0.86	0.51	0.57	**0.83**	0.83	**0.59**	0.76	**0.73**
(e)	2.0	$r./n.$	0.79	0.39	0.45	0.75	0.78	0.51	0.65	0.70
(f)	1.0	$c./o.$	0.92	0.62	0.76	0.77	0.84	0.52	0.88	0.55
(g)	1.0	$r./o.$	0.91	0.62	0.73	0.80	0.84	0.55	0.86	0.61
(h)	1.0	$r./n.$	0.91	0.61	0.70	0.83	0.85	0.59	0.85	0.66

我们首先通过将 k 从 0 调整到 2（a-e），研究面积敏感损失 的影响。所有模型（a-e）均使用从采样相机渲染的视图进行训练和测试。在手动测试集上，我们发现，将 k 从 0 增加到 1.5 时，IoU(C) 提高了 0.1 以上，而 IoU(NC) 仅略有波动。这种不对称变化是由于两个类别的尺寸分布不均衡。考虑到分割

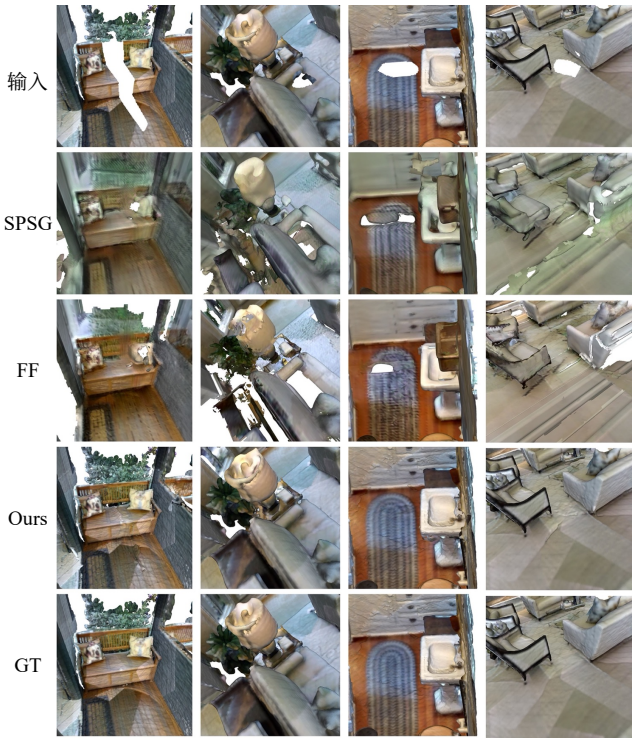

图 4. **三维修复对比**。 我们将本方法与两种三维场景修复（补全）基线方法进行比较（SPSG [12] 和 FF [21]）。第一行展示输入的不完整场景（通过将杂物放置到场景中来形成孔洞），接下来是三种方法的输出，底部则是没有合成孔洞的真实值场景。以往方法受低分辨率限制，而我们的方法可以保持与输入相同的分辨率，并且在补全较大孔洞和较罕见场景方面表现更好。

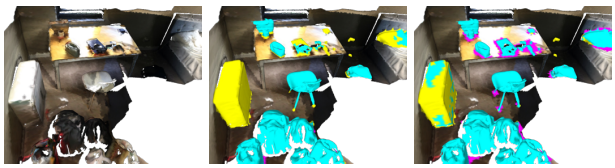

图 5. **尺寸平衡参数对杂物检测影响的可视化** k。 青色标记的区域为真正例，洋红色为假正例，黄色为假负例。

具有噪声标签的小物体很困难，从（a）到（d）杂物交并比的大幅提升证明了我们面积敏感损失的有效性。为了更好地解释这一提升，我们进一步考察了杂物的精确率和召回率。如表的最后两列所示 3 （a-d），较大的 k （在一定范围 0-1.5 内）会显著提高召回率，同时降低精确率。当 k 增至 2 时（表 3 （e）），由于权重过大，各项指标均开始下降。

（a-d）中的上述趋势反映在图 5中。召回率的提高对应于扩大的青色（真阳性）区域和缩小的黄色（假阴性）区域。更大的粉色（假正例）区域意味着精确率下降。借助$k = 1$，模型可以分割桌子上的微小杂物，并完整分割右上方的枕头；而在使用 $k = 0$时，该枕头会被部分遗漏。较低的精确率主要是由杂物周围的假正例（粉色）造成的。由于我们原本也会膨胀预测掩码以去除阴影，因此这并没有预期中那么有害。我们想指出的是，目前遗漏的预测结果都是非常困难的案例。例如，左侧的大箱子与一个小柜子非常相似，而原始类别箱子 本身就是杂物和非杂物的混合类别。右上方两个小的黄色缺失区域是两个小插头，其周围的网格表面非常暗，很难识别出其中的杂物。

在原始测试集中，我们可以观察到，随着 k 增大，IoU(C) 和 IoU(NC) 均会下降。比较测试集 (a) 到 (d) 中 IoU(NC) 的变化，原始测试集中的数值大幅下降（从 0.94 降至 0.86），而人工标注集中的数值保持不变（从 0.83 到 0.83）。这是因为通过 k 增强面积敏感度后，更多杂物会被识别出来，而在原始测试集中，这些杂物可能被误标为非杂物。这也验证了我们先前的主张（第 1 节）：现有数据集中的杂乱标签质量很低。尽管标签质量较低，我们仍观察到与手动测试集相似的趋势，即召回率提高而精确率下降。

我们在表中比较使用渲染图像或捕获图像的效果。3 (c、f、g、h)。所有 (f-h) 实验均在捕获图像上进行测试。可以看到，(g) 使用捕获的相机参数在虚拟渲染结果上训练，其表现与 (f) 使用原始捕获图像训练的表现相当。这表明二维域差异不会影响性能。从 (g) 到 (h)，可以看到，相较于仅使用手持传感器轨迹上的位姿 (g)，更大的相机位姿变化 (h) 带来了性能提升。比较 (c) 和 (h)，二者使用同一模型在 $r./n.$ 视图上训练；可以看到，使用虚拟渲染结果训练的模型，在原始捕获图像上测试 (h) 时，比在其虚拟渲染结果上测试 (c) 时表现更好。这可能是因为在捕获图像上测试更容易，因为所有相机都处于同一条轨迹上。

4.4.2 三维修复

尽管图像修复器能够强大地修复出语义上合理的内容，但始终无法保证跨视角的一致性。在本节中，我们研究应用不同一致性规则所产生的影响，相关结果如图6所示。

对于单帧而言，尽管图像修复内容与剩余像素结合后看起来可能很合理，但图像修复图像可能与原始 RGB 图像不一致。例如，在行 (a)中，尽管修复后的 RGB补全出了一个合理的桌子，但这些原本被椅子遮挡的区域实际上应当是地板。如果我们直接从一个序列重建网格，而其中被遮挡的视图采用了不同的图像修复方式，

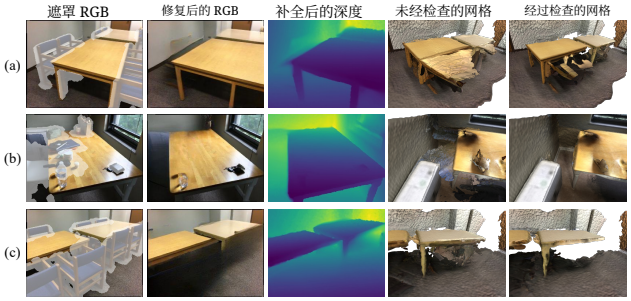

图 6. **一致性规则的必要性及其效果可视化**。原始图像（第一列）分别进行图像修复（第二、三列），这往往会导致合并结果不理想（第四列）。我们的检查使重建结果更加可信（第五列）。三行分别展示了 (a) 单帧剪枝、(b) 跨帧剪枝和 (c) 跨帧投票的效果。

我们将在第四列获得一个网格。一个简单而有效的解决方案是丢弃修复区域的深度值小于原始帧的帧（单帧一致性小节中3.2.2）。

不仅 RGB 图像修复器可能导致重建网格出现伪影，不准确的深度预测也会造成同样的问题。第 (b) 行展示了一个例子：桌子与墙壁之间空间的深度补全很容易产生噪声。第四列中的网格就是未经修正时得到的结果。我们知道这个网格是错误的，因为从类似第三行的视角观察时，应该能够看到墙壁和地板。基于这一跨帧先验，我们采用跨帧一致性剪枝：在一帧中可见的任何未修复区域，在另一帧中也应可见。这一方法能够有效去除许多伪影，并得到更加干净的网格，如第五列所示。

到目前为止，我们已应用所有利用捕获图像的一致性假设。最后一步是检查不同视角下修复区域之间的一致性。例如，桌子下方较暗的修复区域很容易导致深度估计不准确，从而产生第四列中的杂乱伪影。通过采用第 3.2.2 节所述的跨帧投票，不同视角下没有聚类深度的区域将被剔除。

5. 结论

本文研究了从三维场景中检测和移除杂物，同时提供合理填充内容的问题。我们首先将杂物定义为频繁移动物体，并指出现有3D场景分割数据集中标注噪声严重的问题。随后，我们提出了一种用于杂物分割和三维图像修复的解决方案。为了更好地分割形状各异的杂物，我们开发了实例级面积敏感损失，该损失显著提高了召回率，尤其是对于较小的对象。我们的三维修复方法基于将任务分解为几何和颜色（语义）这一关键原则。为此，我们将三维分割投影到RGB-D图像上，以在二维空间中补全颜色和深度。随后，在进行视图一致性投票和剪枝后，利用这些2D图像修复结果进行网格重- 构。就我们所知，这是首项研究三维杂物分割与修复结果相结合问题的工作。尽管我们的方法与基线方法相比表现良好，但仍有几个尚未解决的问题。例如，在进行修复结果之前移除阴影，以及直接在三维空间中解决这一问题的方法，都是值得进一步研究的方向。

致谢 我们谨此感谢英特尔和谷歌对本工作的部分资助。

参考文献

[1] Iro Armeni、Ozan Sener、Amir R Zamir、Helen Jiang、Ioannis Brilakis、Martin Fischer 和 Silvio Savarese。大规模室内空间的3D语义解析。载于 CVPR，第1534–1543页，2016年。1，2[2]Coloma Ballester、Marcelo Bertalmio、Vicent Caselles、Guillermo Sapiro 和 Joan Verdera。通过向量场与灰度级的联合插值进行填充。IEEE图像处理汇刊，10(8)：1200–1211，2001年。2[3] Connelly Barnes、Eli Shechtman、Adam Finkelstein 和 Dan B Goldman。PatchMatch：一种用于结构化图像编辑的随机对应算法。ACM 图形学汇刊（TOG），28(3)：24，2009年。2[4] Angel Chang、Angela Dai、Thomas Funkhouser、Maciej Halber、Matthias Niessner、Manolis Savva、Shuran Song、Andy Zeng 和 Yinda Zhang。Matterport3D：从室内环境中的 RGB-D 数据中学习。arXiv 预印本 arXiv:1709.06158，2017年。1，2，6，12[5] Xinjing Cheng、Peng Wang、Chenye Guan 和 Ruigang Yang。Cspn++：面向深度补全的上下文与资源感知卷积空间传播网络学习。载于 AAAI，第34卷，第10615–10622页，2020年。2，4[6] Paolo Cignoni、Marco Callieri、Massimiliano Corsini、Matteo Dellepiane、Fabio Ganovelli 和 Guido Ranzuglia。MeshLab：一种开源网格处理工具。载于 Vittorio Scarano、Rosario De Chiara 和 Ugo Erra 编，《欧洲计算机图形学协会意大利分会会议》。欧洲计算机图形学协会，2008年。6[7] Paolo Cignoni、Alessandro Muntoni、Guido Ranzuglia 和 Marco Callieri。MeshLab。6[8] Yin Cui、Menglin Jia、Tsung-Yi Lin、Yang Song 和 Serge Belongie。基于样本有效数量的类别平衡损失。载于 CVPR，第9268–9277页，2019年。3，14[9] Angela Dai、Angel X Chang、Manolis Savva、Maciej Halber、Thomas Funkhouser 和 Matthias Nießner。ScanNet：丰富标注的室内场景3D重建。载于 CVPR，第5828–5839页，2017年。1，2，3，5，12

[10] Angela Dai、Christian Diller 和 Matthias Nießner。Sg-nn：用于 RGB-D 扫描自监督场景补全的稀疏生成神经网络。载于 CVPR，2020年。2[11] Angela Dai、Daniel Ritchie、Martin Bokeloh、Scott Reed、J¨urgen Sturm 和 Matthias Nießner。ScanComplete：面向 3D 扫描的大规模场景补全与语义分割。载于 CVPR，2018年。2[12] Angela Dai、Yawar Siddiqui、Justus Thies、Julien Valentin 和 Matthias Nießner。Spsg：从 RGB-D 扫描中进行自监督光度场景生成。载于 CVPR，第1747–1756页，2021年。2，3，4，6，8，15[13] Alexei A Efros 和 William T Freeman。用于纹理合成与迁移的图像拼接。载于 SIGGRAPH 论文集，第341–346页，2001年。2[14] Alexei A Efros 和 Thomas K Leung。通过非参数采样进行纹理合成。载于 ICCV，第2卷，第1033–1038页。IEEE，1999年。2[15] David Eigen 和 Rob Fergus。通过通用的多尺度卷积架构预测深度、表面法线和语义标签。载于 ICCV，第2650–2658页，2015年。3，14[16] Youichi Horry、Ken-Ichi Anjyo 和 Kiyoshi Arai。走进画面：使用蜘蛛网状网格界面从单张图像生成动画。载于 SIGGRAPH 论文集，第225–232页，1997年。2[17] Mu Hu、Shuling Wang、Bin Li、Shiyu Ning、Li Fan 和 Xiaojin Gong。PENet：迈向精准高效的图像引导深度补全。载于 ICRA，第13656–13662页。IEEE，2021年。2，4[18] Wenbo Hu、Hengshuang Zhao、Li Jiang、Jiaya Jia 和 Tien-Tsin Wong。用于跨维度场景理解的双向投影网络。载于 CVPR，2021年。1，2，3，4，6，13，14[19] Shruti Jadon。语义分割损失函数综述。载于 CIBCB，第1–7页。IEEE，2020年。3[20] Varun Jampani、Huiwen Chang、Kyle Sargent、Abhishek Kar、Richard Tucker、Michael Krainin、Dominik Kaeser、William T Freeman、David Salesin、Brian Curless 等。SLIDE：结合软分层和深度感知图像修复的单图像三维摄影。载于 ICCV，第12518–12527页，2021年。2[21] Ru-Fen Jheng、Tsung-Han Wu、Jia-Fong Yeh 和 Winston H Hsu。使用双流生成对抗网络进行自由形式三维场景修复。BMVC，2022年。2，3，4，6，8，12，15[22] Michael Kazhdan、Matthew Bolitho 和 Hugues Hoppe。泊松表面重建。载于第四届 Eurographics 几何处理研讨会论文集，第7卷，2006年。2，3，5，6，15[23] Johannes Kopf、Kevin Matzen、Suhib Alsisan、Ocean Quigley、Francis Ge、Yangming Chong、Josh Patterson、Jan-Michael Frahm、Shu Wu、Matthew Yu 等。单次三维摄影。ACM 图形学汇刊（TOG），39(4)：76–1，2020年。2[24] Abhijit Kundu、Xiaoqi Yin、Alireza Fathi、David Ross、Brian Brewington、Thomas Funkhouser 和 Caroline Pantofaru。用于3D语义分割的虚拟多视图融合。载于 ECCV，第518–535页。Springer，2020年。1，2，4

[25] Feiran Li、Gustavo Alfonso Garcia Ricardez、Jun Takamatsu 和 Tsukasa Ogasawara。用于 RGB-D 序列的多视图图像修复。载于 3DV，第 464–473 页。IEEE，2018年。2[26] Jingyuan Li、Ning Wang、Lefei Zhang、Bo Du 和 Dacheng Tao。用于图像修复的循环特征推理。载于 CVPR，第 7760–7768 页，2020 年。2[27] Wenbo Li 、 Zhe Lin 、 Kun Zhou、Lu Qi、Yi Wang 和 Jiaya Jia。MAT：用于大孔洞图像修复的掩码感知变换器。载于 CVPR，第 10758–10768 页，2022年。2[28] Tsung-Yi Lin、Priya Goyal、Ross Girshick、Kaiming He 和 Piotr Doll´ar。用于密集目标检测的焦点损失。载于 ICCV，第 2980–2988 页，2017年。14[29] Adrian Lopez-Rodriguez、Benjamin Busam 和 Krystian Mikolajczyk。Project to adapt：基于含噪稀疏传感器数据的深度补全域适应。载于 ACCV，2020年。2，4[30] Fangchang Ma、Guilherme Venturelli Cavalheiro 和 Sertac Karaman。自监督稀疏到稠密：基于激光雷达和单目相机的自监督深度补全。载于 ICRA，第 3288–3295 页。IEEE，2019年。2，4[31] Ashkan Mirzaei、Tristan Aumentado-Armstrong、Konstantinos G Derpanis、Jonathan Kelly、Marcus A Brubaker、Igor Gilitschenski 和 Alex Levinshtein。SPIn-NeRF：利用神经辐射场进行多视图分割与感知图像修复。载于 CVPR，第 20669–20679 页，2023年。3[32] Himangi Mittal、Brian Okorn、Arpit Jangid 和 David Held。通过图像修复进行自监督点云补全。arXiv 预印本 arXiv:2111.10701，2021年。2[33] Shohei Mori、Jan Herling、Wolfgang Broll、Norihiko Kawai、Hideo Saito、Dieter Schmalstieg 和 Denis Kalkofen。3D PixMix：三维环境中的图像修复。载于 2018年 IEEE混合与增强现实国际研讨会附属会议（ISMAR-Adjunct），第 1–2 页。IEEE，2018年。2[34] Kamyar Nazeri、Eric Ng、Tony Joseph、Faisal Z Qureshi 和 Mehran Ebrahimi。EdgeConnect：采用对抗边缘学习的生成式图像修复。arXiv 预印本 arXiv:1901.00212，2019年。2[35] Alexey Nekrasov、Jonas Schult、Or Litany、Bastian Leibe 和 Francis Engelmann。Mix3D：用于 3D场景的上下文外数据增强。载于 3DV，第 116–125 页。IEEE，2021年。1，2[36] Jinsun Park、Kyungdon Joo、Zhe Hu、Chi-Kuei Liu 和 In So Kweon。用于深度补全的非局部空间传播网络。载于 ECCV，2020年。2，4，14[37] Deepak Pathak、Philipp Krahenbuhl、Jeff Donahue、Trevor Darrell 和 Alexei A Efros。上下文编码器：通过图像修复进行特征学习。载于 CVPR，第 2536–2544 页，2016年。2[38] Songyou Peng、Chiyu “Max” Jiang、Yiyi Liao、Michael Niemeyer、Marc Pollefeys 和 Andreas Geiger。形状即点：一种可微泊松求解器。载于 NeurIPS，2021年。2[39] Julien Philip 和 George Drettakis。用于大型场景基于图像渲染的基于平面的多视图图像修复。载于 ACM SIGGRAPH交互式三维图形与游戏研讨会论文集，第 1–11 页，2018年。2[40] Yurui Ren、Xiaoming Yu、Ruonan Zhang、Thomas H Li、Shan Liu 和 Ge Li。StructureFlow：通过……进行图像修复

结构感知外观流。在ICCV，第 181 190 页，2019年 2 – 。
[41] David Rozenberszki、OrLitany 和 AngelaDai。野外基于语言的室内 3D语义分割。在 ECCV，2022年。 1、2、6[42] ShankarSetty和UmaMudenagudi。使用度量张量和克里斯托费尔符号对点云进行基于示例的 3D 图像修复。Machine VisionandApplications，29 343，2018年 2329 2 ( ): –。 [43] Meng-Li Shih、Shih-Yang Su、JohannesKopf和 Jia-Bin Huang。使用上下文感知的分层深度图像修复进行3D摄影。在 CVPR，第 8028–8038 页，2020年。 2[44] Roman Suvorov、Elizaveta Logacheva、Anton Mashikhin、Anastasia Remizova、Arsenii Ashukha、Aleksei Silvestrov、Naejin Kong、Harshith Goka、Kiwoong Park和 VictorLempitsky。使用傅里叶卷积进行分辨率鲁棒的大掩码图像修复。在WACV，第 2149–2159 页，2022年。 2、4[45] Theo Thonat、Eli Shechtman、SylvainParis和GeorgeDrettakis。用于基于图像的场景编辑和渲染的多视图图像修复。在
3 351 359 2016 2 DV，第 –
[46] 页。 IEEE，。 Ziyu Wan、JingboZhang、DongdongChen 和 Jing Liao。使用变换器进行高保真多样化图像补全。在 ICCV，第
4692 4701, 2021 2 – 页。 XinyingWang、
[47] Dikai Xu和Fangming Gu。基于 3D 深度卷积生成对抗网络的 3 D 模型图像修复。IEEEAccess，8:170355–170363，2020年。 2 [48] SilvanWeder、GuillermoGarcia-Hernando、Aron Monsz-part、Marc Pollefeys、Gabriel J Brostow、Michael Firman 和 SaraVicente。从神经辐射场中移除对象。在 CVPR，第
16528 16538, 2023 3 – 页。Alex
[49] Wong和Stefano Soatto。使用校准反投影层进行无监督深度补全。在ICCV，第 12747–12756 页，2021年。 2、4[50] Hailong Yan、Wenqi Wu、Zhenghua Deng、Junjian Huang、Zhizhang Li和 Luting Zhang。基于已知区域边界的 3D 重建图像修复。 Mathematics，
10 15 2761, 2022 2 ( ): 。
[51] YanchaoYang、Alex Wong和StefanoSoatto。来自单幅图像和稀疏距离的稠密深度后验（ddp）。在 CVPR，第 3353–3362 页，2019年。 2、4[52] Jiahui Yu、Zhe Lin、Jimei Yang、XiaohuiShen、XinLu和 Thomas SHuang。使用门控卷积进行自由形状图像修复。在 ICCV，第 4471–4480 页，2019年。 2、7[53] HaoranZhang、Zhenzhen Hu、ChangzhiLuo、WangmengZuo和 Meng Wang。使用渐进式生成网络进行语义图像修复。在第 26 届 ACM国际多媒体会议论文集，第 1939–1947 页，2018年。2[54] Shengyu Zhao、Jonathan Cui、Yilun Sheng、Yue Dong、XiaoLiang、EricI Chang 和YanXu。通过协同调制生成对抗网络进行大规模图像补全。arXiv 预印本 arXiv:
2103 10428, 2021 2 。。 Chuanxia Zheng、
[55] Tat-Jen Cham 和 Jianfei Cai。多样化图像补全。在 CVPR，第
1438 1447, 2019 2, 7 – 页。
[56] Xizhou Zhu、HanHu、StephenLin 和 Jifeng Dai。可变形卷积网络 v2：更强的可变形性，更好的结果。在 CVPR，第 9308–9316 页，2019年。 2、14

在本补充材料中，我们首先详细介绍用于评估三维分割和三维图像修复的数据集，见第 A 节。然后，我们在第B节中介绍更多实现细节。最后，我们在第 C 节中展示我们方法的更多结果。

A. 数据集

A.1. 杂物分割

我们从 ScanNet测试集中选择了八个具有代表性的场景，以尽可能覆盖数据集中的多种场景类型。我们使用 ScanNet的标注工具 [9] 在杂物上标注顶点。在图7中，我们展示了八个场景的鸟瞰图，这些场景均由我们手动标注，用于评估杂乱物体分割模型。具体而言，手动测试集涵盖了范围广泛的整洁程度，从仅有很少常见杂物的房间到杂物极多的房间。涵盖的房间类型包括书房、会议室、卧室、浴室、客厅、工作室和手工艺室。正如第三列所示，在原始 ScanNet 标签中，一些杂物未被标注（例如，第二行，圆桌上的物体），而一些物体被错误标注（例如，最后一行，深蓝色衣柜中的杂物）。相比之下，我们手动标注的数据集正确地标注了场景中的所有杂物。标注每个场景所需的时间从30分钟（仅适用于极其整洁的场景）到数小时不等。这也表明，要为包含数千个场景的数据集精确标注杂物/非杂物标签是不切实际的。

在所有实验中，我们每隔 5 帧从原始采集序列中采样一次，以减少冗余并提高效率。为了渲染用于杂物分割的虚拟视图，我们使用内参固定的相机。我们将相机随机放置在距网格表面 2 至 5 米处，朝向场景渲染视图，用于训练。

A.2. 三维修复

用于定量评估三维修复的指标（主文表 2）需要真实标注，但现有数据集中都没有此类标注。因此，我们构建了一个合成评估集。我们使用 Matterport3D 官方测试集 [4] ，其中原本包含 391 个房间。对于每个房间，我们找出属于非杂物类别且具有归一化法线的区域 $n = (n_x,\ n_y,\ n_z)$，其中 $|n_z| = max(|n_x|,\ |n_y|,\ |n_z|)$ (z指向上方）。与此同时，我们从场景中移除属于纯杂物类别的实例，并将这些实例放置在已识别的非杂物区域上。我们计算放置的杂物的凸包，并移除房间内被该凸包包围的任何表面，从而生成孔洞。换言之，原始房间是三维修复后的真实标注房间，而引入孔洞后的房间则是各种三维修复方法的输入。

表 4.**运行时间分析**。我们根据约有 200 帧、且来自原始扫描数据的输入网格约含 150，000 个顶点的场景，计算平均耗时。所有图形处理器均为配备 12 GB 显存的 NVIDIA TITAN X。可以看到，处理一个包含 200 帧的 RGB-D 序列耗时略超 30 分钟，其中大部分时间花在一致性检查和表面重建上。

Step	Time	设备	输出尺寸
杂物分割	2s	GPU	150,000 点
掩码投影	30s	GPU	200 张图像
图像修复	50s	GPU	200 张图像 320×240
深度补全	80s	GPU	200 张图像 304×228
一致性检查	18 分钟	GPU	200 张图像 304×228
表面重建	11 分钟	CPU	1,500,000 个顶点

两种基线方法都以 TSDF 表示作为输入和输出。为生成输入，我们将凸包范围内体素的 TSDF 值和颜色值设为掩码。这与 FF [21] 在其方法中生成自由形状孔洞的方式非常相似。

对于房间中的每个孔洞，我们从数据集中找到一个视角，使孔洞尽可能靠近图像中心。我们还会找到与中心视角至少有 1000 像素重叠的其他视角。我们将杂物投影到上述所有视角中，在 RGB-D 数据上生成孔洞。随后，我们使用这些视角运行 RGBD修复和网格重建。

下面我们说明在获得图像修复后的三维结果（无论来自我们的方法还是基线方法）后，如何计算三维和二维指标。为评估所提方法的三维性能，我们使用原始（未遮挡）视图，通过 PSR 重建真实网格，并计算其与我们修复后的网格之间的倒角距离。为评估基线方法的三维性能，我们计算由预测的 TSDF 生成的移动立方体网格与由真实 TSDF 生成的移动立方体网格之间的倒角距离。与将 PSR 重建的网格用作基线方法的真实网格相比，这种评估更加公平（将 PSR 的结果用作真实网格，得到的结果会比使用 TSDF 生成的移动立方体网格更差，因此我们不将 PSR 用作基线方法的真实网格）。为评估二维性能，我们从居中视图进行渲染，并计算真实渲染结果（不含被移除的杂物）与我们的方法或基线方法生成的渲染结果之间的指标。

B. 实现细节

我们在表4 中展示了一个200 帧序列中每个步骤的平均耗时。具体而言，我们根据大约包含 200 帧的场景以及一个 150,000 顶点的输入网格计算平均耗时，该输入网格来自原始扫描数据。所有图形处理器均为配备12 GB 内存的 NVIDIA TITANX。我们可以看到，处理一个 200帧的 RGB-D序列需要略多于 30 分钟，其中大部分时间花费在一致性检查和表面重建上。

我们通过实验发现，两个类别的交叉熵损失

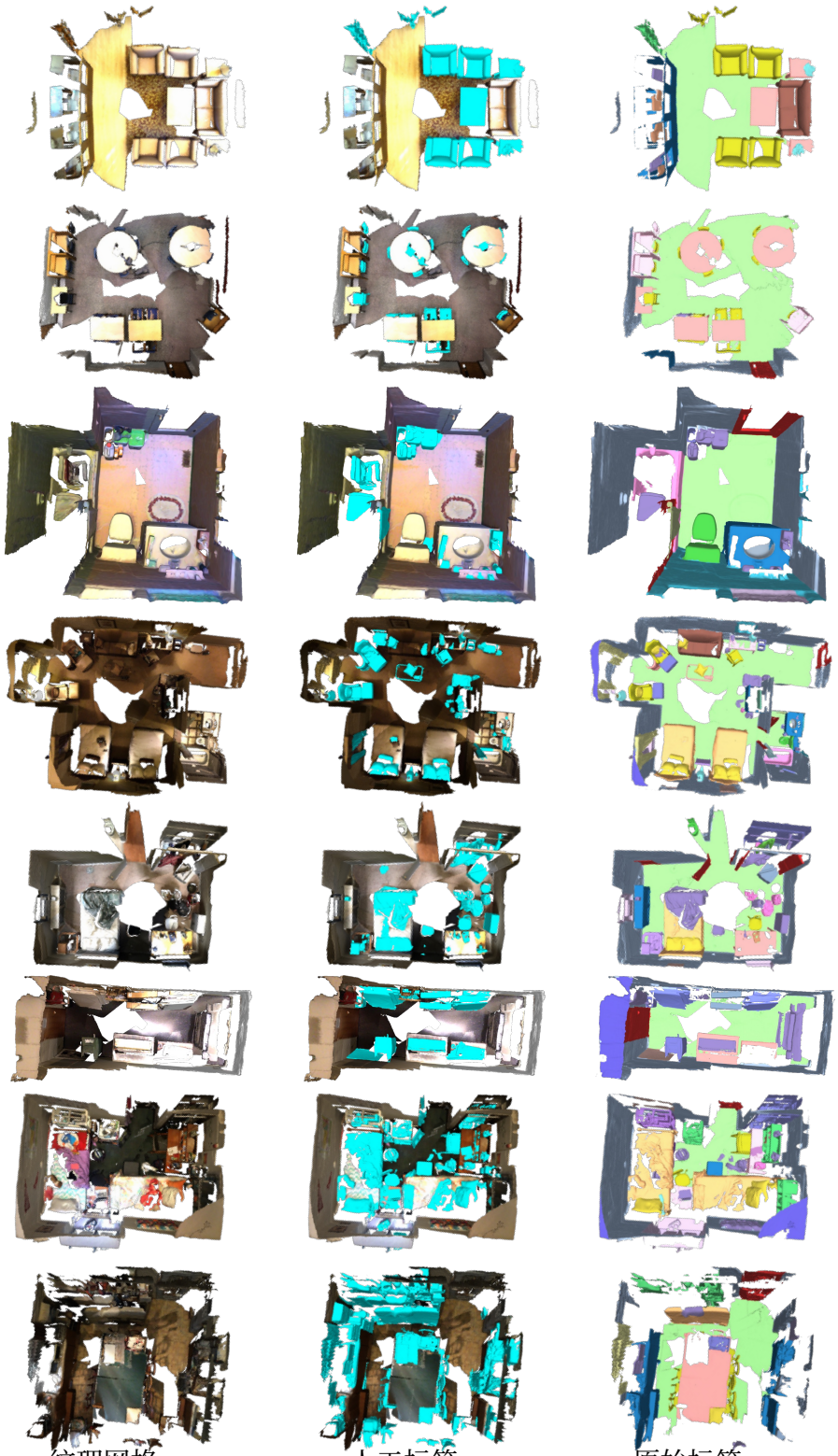

图7。**杂物分割手工标注的评估数据集**。我们展示了全部八个场景的鸟瞰图（经过缩放、旋转并移除了天花板表面，以获得更好的视图），这些场景由我们手动标注，用于评估我们方法的杂乱物体分割模型和BPNet [18] 基线。

效果优于二元交叉熵损失。

B.1. 杂乱定义

我们的重点是提出一种基于共享属性（例如，杂物之间的共享属性）而非常见基准语义类别的新型场景分割方法。因此，明确的杂物定义对于检验我们的方法至关重要，但其具体形式可以非常灵活。

我们在实验中采用的定义如下：如果一个未安装物体在 3 个月内的任意时刻发生移动，且移动前后边界框的交并比为 < 0.5，其概率为 $> 95\%$，则该物体被视为杂物。需要指出的是，尽管运行实验需要一个明确的定义，但我们提出的杂物分割和三维修复技术具有通用性，并非专门针对某一种定义。例如，可以轻松地将当前的杂物定义扩展为采用可变的时间长度（例如，1 个月或 9 个月），而不会影响所提方法的有效性。

B.2. 杂物分割

模型使用 PyTorch 实现，在两块配备12GB 内存的 NVIDIA TITAN X 上以批量大小 16 进行训练。遵循 BPNet [18], 我们使用随机梯度下降优化器，基础学习率为0.01，并采用幂设为 0.9的多项式学习率调度器。动量和权重衰减分别设置为 0.9 和0.0001。模型总共训练 100 个训练轮次。

测试期间，我们在同一网格上多次运行推理（每次使用不同的相机视角），直到用完所有相机视角。我们累积模型预测的类别概率，以生成最终的 3D 分割结果。

将三维分割掩码投影到RGB-D图像后，我们进一步对二维掩码进行6次迭代（像素）膨胀，以应对三维重-建与二维采集图像之间的错位。最终的掩码应用于颜色和深度图像，以移除包含杂物的区域。

B.3. 深度补全

如正文所述，我们采用 NLSPN [36]进行非局部深度传播，以实现图像引导的深度补全。该模型最初是为稀疏深度补全任务设计的，旨在根据数百个采样像素预测完整的深度图。该方法的关键组件之一是可变形卷积 [56], 其有效采样场可以覆盖整幅图像。因此，尽管该方法最初的目标不同，但它仍适用于我们的用途：补全带有（可能较大）孔洞的稠密图。我们使用带有合成孔洞的深度图对模型进行重新训练，以实现深度补全。下面将详细说明我们如何准备训练数据。

表5.**分割损失的消融实验**。 我们将面积敏感损失与原始交叉熵（CE）、平衡交叉熵（BCE） [8], 中位频率损失（MF） [15], 和焦点损失（FL） [28]进行比较。注：对于其他方法，我们按照原论文建议的最佳实践设置超参数（BCE 中为 β，FL 中为 γ）。

方法	手动测试集（干净）		
	交并比(NC)	交并比(C)	平均交并比
CE	0.83	0.48	0.66
BCE	0.74	0.38	0.56
FL	0.83	0.45	0.64
MF	**0.84**	0.54	0.69
Ours	**0.84**	**0.58**	**0.71**

我们使用经过修改的采集深度数据。由于杂物移除前后都没有真实深度图，我们需要自行合成训练数据。为此，我们从一张采集到的深度图 d_1开始。我们假设m_1 是 d_1的粗略杂物/非杂物分组掩码（正文第 3.2.1 节）。我们使用来自第二个视角的杂物掩码 m_2, 遮盖 d_1上的区域。同时，我们确保 m_1 中原先被标记为杂物的区域不会被遮盖。通过从另一个视角复制并粘贴m_2，我们确保训练期间的掩码具有真实的杂物边界。由于不会遮盖 m_1中的杂物，我们可以保证深度补全模型无需在孔洞区域内臆造杂物。这正是深度补全模型所期望的行为。我们注意到，由于杂物/非杂物分组非常嘈杂，按上述方法生成的训练数据仍然含有噪声。不过，我们通过实验发现，以这种方式训练的模型，效果已经远胜于使用在稀疏到稠密深度补全任务上训练的预训练模型权重。

模型使用 PyTorch 实现，并在一块配备 12 GB 内存的 NVIDIA TITAN X 上以批量大小 12 进行训练。遵循 NLSPN [36], , 我们使用 Adam 优化器，$\beta_1 = 0.9$， $\beta_2 = 0.999$ 初始学习率为 0.001。模型总共训练 20 轮。我们将每次前向传播的传播时间设为 18。

C. 更多结果

C.1. 分割损失的消融实验

表 5展示了比较不同旨在应对不平衡问题的损失函数的消融实验。我们针对每个实例的面积敏感损失表现最佳。BCE [8] 和FL [28]在杂物分割区域上的表现不如 CE 基线。MF [15] 通过根据累积类别表面积对类别进行重新加权，相较于 CE 有所改进，但仍不如我们的方法。由于杂物通常由许多较小的对象实例组成，因此实例级加权更适合此任务。

表 6. **一致性检查消融实验**。sp、cp、cv 分别表示单帧剪枝、跨帧剪枝和跨帧投票。

ID	sp	cp	cv	L1(↓)	L2(↓)	峰值信噪比(↑)	结构相似性(↑)	LPIPS(↓)	倒角距离(cm)(↓)
a				0.034	0.075	21.820	0.860	0.201	9.013
b	✓			0.029	0.051	22.403	0.873	0.189	5.245
c	✓	✓		0.023	0.043	22.965	0.880	0.171	2.612
d	✓	✓	✓	0.019	0.035	23.820	0.891	0.150	0.931

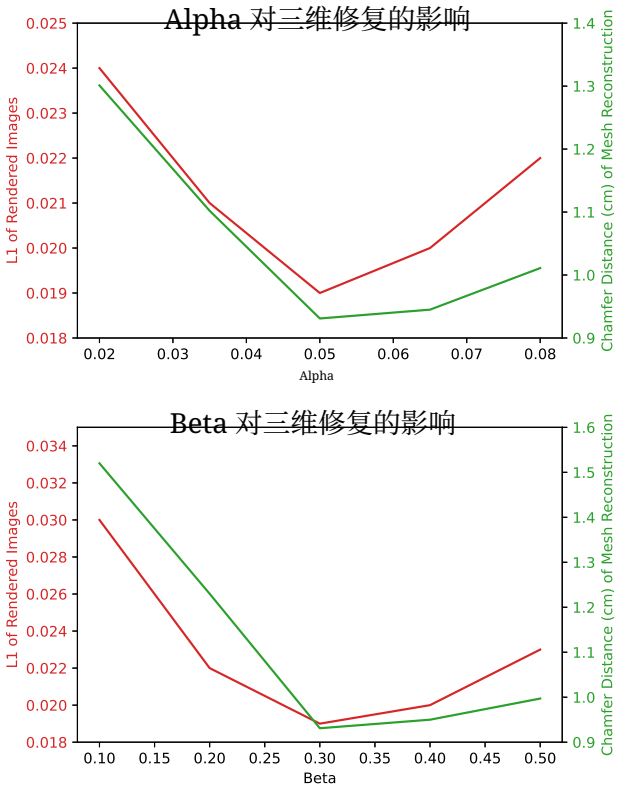

图 8。 **在三维几何和渲染图像上对 α、 β 进行消融实验**。

C.2. 一致性检查的消融实验

表。6研究了每项一致性检查对 100 个 Matterport3D 测试场景重建的贡献（第 A.2节）。每项一致性检查都能改善 3D 修补，尤其是网格重建质量。

C.3. 对 α， β进行消融实验

图 8 研究了不同 α 和 β 对 3D 场景修复的影响。β 数值相对较高，因为同一区域的重叠视图有限；较低的 β 会增加对噪声的敏感度。

C.4. 定性结果

在图 9中，我们展示了物体移除和三维修复完整流程的更多定性结果，并与三维修复基线方法进行了比较。布局与论文正文相同。除了输入网格的纹理和几何外，我们首先展示预测的 3D 杂乱物体分割。随后展示最终修复后的网格的纹理和几何。最后两列中，我们将其与泊松表面重建（PSR）的几何结果 [22]以及孔洞填充算法的结果进行比较。为确保比较公平，输入网格（前两列）也采用论文正文第 3.3.3 节所述的相同泊松表面重建设置进行重建。用于可视化分割结果的网格（第三行）来自原始 ScanNet 数据集。与根据第 3.3.3 节重建的网格相比，这是一个经过大幅平滑和简化的版本。

在图 10 和图11中，我们展示了三维修复与两种基线方法 SPSG [12] 和FF [21]的更多结果。可以看出，与采用体表示的以往工作相比，我们的方法输出了更高分辨率的结果。与基线方法相比，所提出的方法还能够以更加连贯的纹理和几何填补孔洞。



图 9。**自动杂物移除和三维修复的更多结果**。每行展示一个输入三维场景的渲染视图，以及我们的杂物分割结果和场景中颜色与几何均经图像修复后的渲染视图。我们将后者与移除杂物并使用泊松表面重建（PSR）或基于三角剖分的孔洞填充算法填补由此产生的孔洞进行比较。

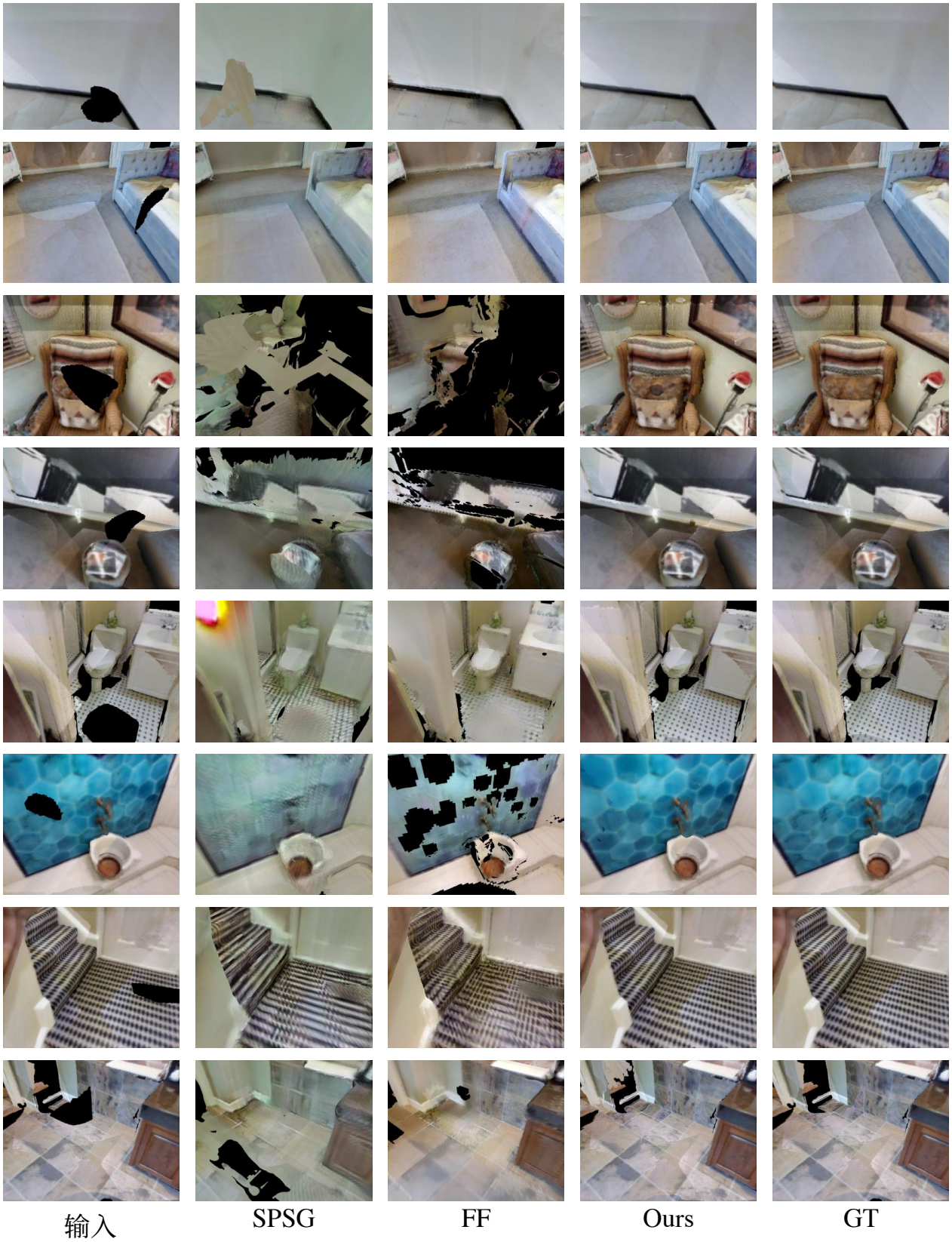

图 10。三**维修复的更多结果**。第一列展示输入三维场景的渲染视图，第二列和第三列分别展示基线 SPSG 和 FF 的结果。第四列展示所提方法的结果。最后一列是真实网格。

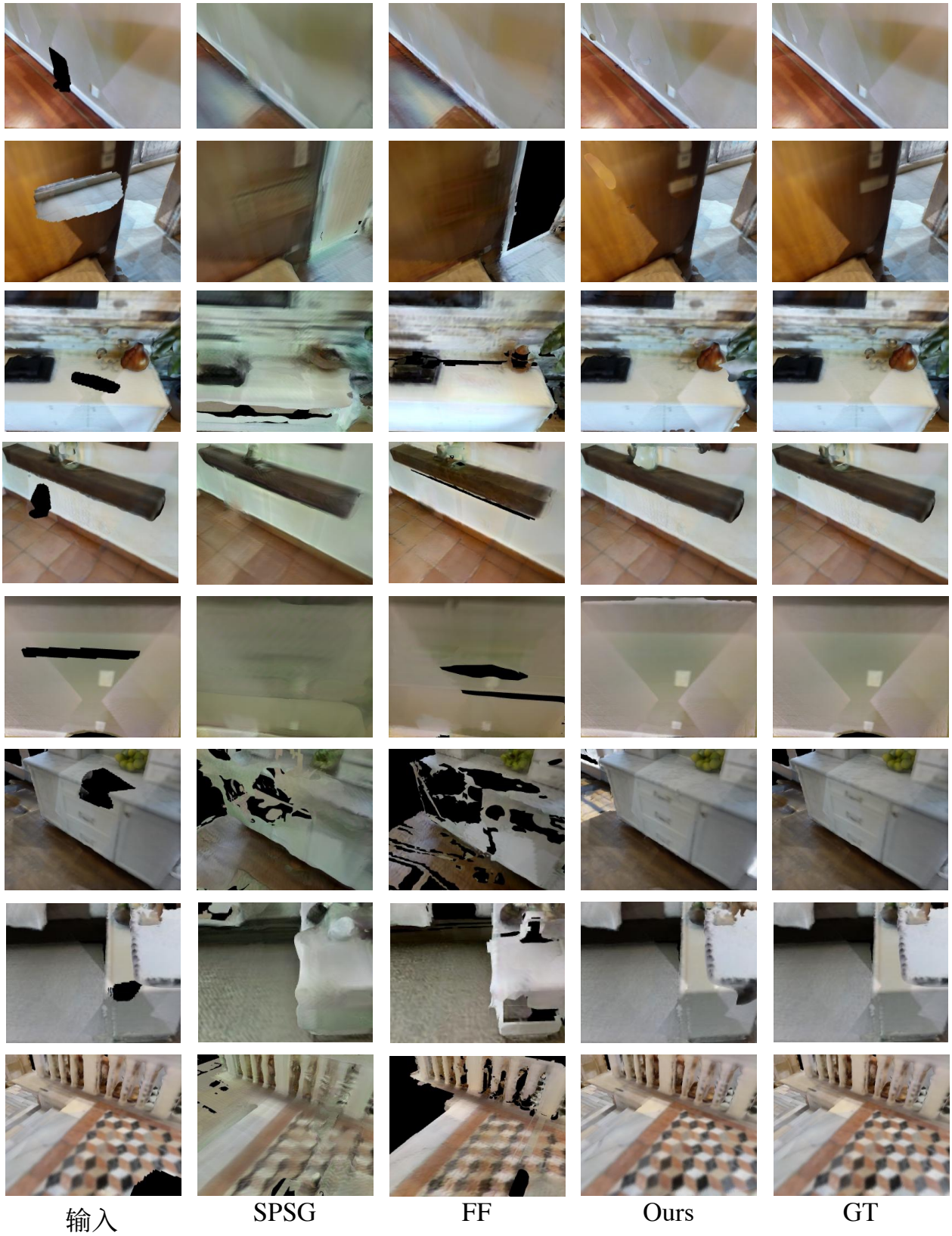

图 11。三**维修复的补充结果**。第一列展示输入三维场景的渲染视图，第二列和第三列分别是基线 SPSG和 FF 的结果。第四列是所提出方法的结果。最后一列是真实网格。