**睿尔曼新零售机器人补充方案**

**技术补充方案（v3.0）**

目录

[睿尔曼新零售机器人补充方案 1](#_Toc18711)

[1. 目标检测识别技术流程 3](#_Toc4790)

[1.1. 基于EfficientSAM的目标识别网络 3](#_Toc22157)

[1.1.1. 数据集采集 3](#_Toc28736)

[1.1.2. 实验方案 6](#_Toc6644)

[2. 位姿估计及位姿跟踪技术数据采集 6](#_Toc1318)

[2.1. 基于SwinDRNet的深度补全网络 6](#_Toc25750)

[2.2. 基于Foundationpose的位姿估计网络 7](#_Toc5472)

[2.3. （补）基于SAM6D的位姿估计网络 7](#_Toc16159)

[2.3.1. 数据集采集 10](#_Toc20795)

[2.3.2. 实验方案 12](#_Toc25036)

[3. 地图感知和路径规划规划 19](#_Toc29978)

[3.1. 超市场景离线建图和建模 19](#_Toc28185)

[3.2. 货架及货架上取货补货点位建模 19](#_Toc30518)

[3.3. 小车避障方案 19](#_Toc16350)

[4. 常见物体的实时抓取估计技术流程 19](#_Toc28534)

[4.1. 基于CenterGrasp的抓取网络 20](#_Toc22882)

[4.1.1. 数据集采集 20](#_Toc1017)

[4.1.2. 实验方案 22](#_Toc5075)

[4.2. 基于物体表面点云的吸取网络 22](#_Toc6995)

[4.2.1. 实验方案 22](#_Toc28122)

[5. 仿真验证 22](#_Toc20642)

[5.1. 基于ros的仿真平台验证 22](#_Toc6373)

[5.1.1. 实验方案 22](#_Toc12089)

数据集量级很大程度上可以影响模型的性能的上限，例如EfficientSAM在SA-1B经过微调训练后，本身具备很好的泛化性能。但本项目基于超市场景，与SA-1B的数据分布存在较大差异，进行数据采集并微调可以取得可预见的收益。

基于 NVIDIA Omniverse 构建的 NVIDIA Isaac Sim 是一个完全可扩展的参考应用，可用于生成逼真的合成数据，大大扩展和方便数据采集工作。



图1： NVIDIA Isaac Sim仿真图片示例

# 目标检测识别技术流程

## 基于EfficientSAM的目标识别网络

## 数据集采集

需要对每个商品进行采集标注，尽可能模拟实际场景进行标注。一方面，真实场景中遮挡堆叠的情况较多，过多的标注信息会导致模型学习到我们不需要的特征；另一方面，模型不需要一次性输出全部的分类结果，只需要输出位于表面清晰可见的商品及mask用于下一步6D位姿和操作使用。综合以上，提出大致数据要求：

1. 尽量模拟真实环境光线视角，采集rgb图片；
2. 图片标注未被遮挡部分的mask和分类标签；
3. 也需对单独物体进行采集标注；
4. 数据集文件格式参照coco，SA-1B；

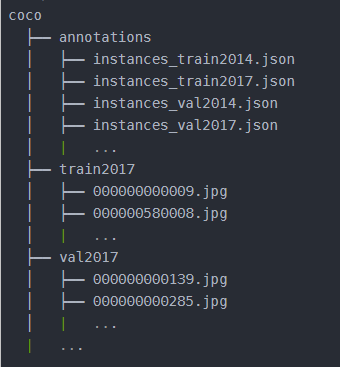


图2：coco数据集文件结构示例

1. 初期每种商品带有标注的图像至少300-500张，后续根据实验结果再确定图片数量。

具体的说，所有COCO目标检测框和mask信息都放在json文件中，json文件里面存放标注相关的字典。制作数据集需要按照标准格式以减少后期数据格式与模型不统一的问题。

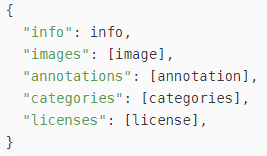


图3：json文件中的字典

info和licenses字段是不需要的，只需要存放images，annotations和categories即可。

其中，Images是一个字典的列表，储存图像的文件名，高度和id，id是图像的编号，与annotations中的id一一且唯一对应。有多少张图片，该images列表就有多少个字典。

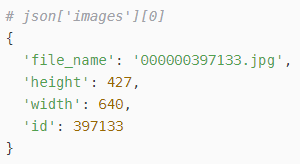


图4：images列表中的字典

categories表示所有的类别，有多少类就定义多少，类别的id从1开始，0为背景。格式如下：

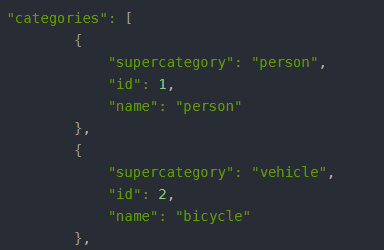


图5：categories列表示例

annotations中是实例分割的标注，格式如下：

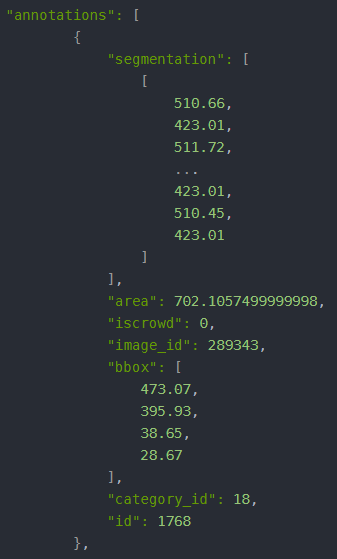


图6：annotations列表示例

其中：

segmentation表示分割标签；

area表示面积；

iscrowd表示是否多个目标；

image\_id与images中的id一一对应；

Bbox表示目标框；

Category\_id表示类别；

Id表示标注框的序号。

根据上述格式制作，即可满足本项目目标识别技术所需的微调数据集。

## 实验方案

使用EfficientSAM进行目标检测识别；

## 模型训练方面：

1. 综述：

EfficientSAM使用通过SIAM预训练好的轻量级ViT图形编码器和mask解码器来重构SAM，并在SA-1B上对模型进行微调，提高了SAM模型的效率同时也获得了高性能。具体结构由图1所示：

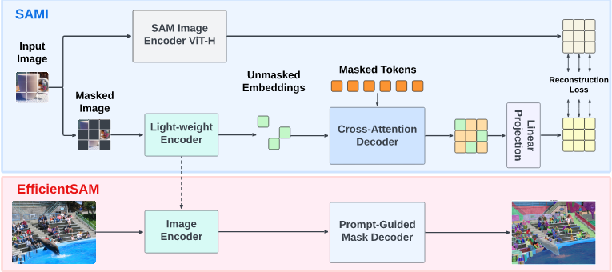


图 1：EfficientSAM模型训练结构

1. 模型输入：rgb图像，图片标注未被遮挡部分的mask和分类标签，单独物体采集标注（每种商品带有标注的图像至少300-500张）
2. 模型微调：我们子啊imagenet-1k上使用100个epochs进行预训练
3. 模型输出：目标物体的位置，轮廓，类别信息

大致流程：数据集导入**——>**ImageNet上的SAMI预训练**——>**SA-1B上的SAM微调**——>**得到模型**——>**利用模型输出目标信息

对于本项目而言，数据集使用的是基于数据集采集要求的数据，通过该数据集建立的进行SAMI预训练与SA-1B上的SAM微调可以得到更适合该项目要求的模型，此时，我们可以运用全量更新参数的方法适配我们的数据集。

1. 具体实验优化过程：

预训练数据集 -> 模型预训练实施细则 -> 三类数据集分别进行不同的图像训练 -> 分割实验 -> 模型主干 -> 微调 -> 检测和实例分割任务

## 各阶段情况:

1. 预训练数据集  
    利用 ImageNet - 1K 训练集，采用 SAMI（掩蔽图像预训练生成方法）进行预训练。在掩蔽图像预训练之后，撤除掉标签信息。在预训练 ViT - Tiny、ViT - Small 和 ViT - Base 模型时，使用 SAM ViT - H 图像编码器来生成重建特征，从而成功构建适配本项目的数据集。
2. 模型预训练实施细则  
   ViT 模型经过预训练，以均方误差（MSE）（用于重建）作为损失度指标衡量模型拟合程度。
3. 优化器：计划使用 AdamW 优化器。
4. 学习率：2.4e - 3。
5. 权重衰减：0.05。
6. 预热迭代：40 次。
7. 学习率更新：利用余弦学习率衰减。
8. 数据增强操作：

* 随机调整大小裁剪至 224\*224 频率。
* 通过随机水平翻转和标准化来增加数据。
  + 适当调整掩码比以达最佳效果。

1. 3三类数据集图像训练  
    考虑三个基准数据集和几个代表性的视觉任务，以证明 SAMI 的优越性：  
    ① 在第一个基准数据集（ImageNET）上：

* 训练集：使用 120 万张图像。
* 测试集：使用 5 万张图像进行图像分类训练。

② 在第二个数据集（coco 数据集）上：

* 训练集：使用 11.9 万张图像。
* 测试集：使用 5 千张图像进行对象检测和实例分割。

③ 在第三个数据集（ADE 数据集）上：

* 训练集：使用 2 万张图像进行语义分割。
* 验证集：2 千张图像。
* 测试集：3 千张图像。

1. 分割  
    在 SA - 1B 的数据集上，使用 1100 万高分辨率图像（超过 10 亿个掩码）对 SAM 的预训练轻量级图像编码器进行微调，并在 COCO 和 LVIS 上测试 EfficientSAM 的交互式实例分割和零激发实例分割能力。
2. 模型主干  
    由于 SAMI 解码器存在缺点，丢弃 SAMI 的解码器但保留编码器作为编码器主干，用于提取不同任务的特征（如 MAE 预训练方法中所述）。将预训练好的 ViT 编码器主干应用于不同任务，包括：

* 用于分类的 ViT。
* 用于检测和实例分割的 ViTDet。
* 用于分割任何内容的 SAM。

1. 微调设置

* 优化器：使用 AdamW 优化器（β1 = 0.9，β2 = 0.999，权重衰减 0.05）。
* 硬件资源：使用 32 个 V100 的 GPUs。
* 分类任务：迭代 100 次，将 ViTSmall 和 Vit - Base 的分层学习率衰减因子设置为 0.75。
* 数据增强：采用 RandAugment。
* 标签平滑：设置为 0.1。
* 混合：设置为 0.8。

1. 检测和实例分割任务

①检测任务：

* 遵循 ViTDet 主干，将 ViT 编码器主干适配为简单的特征金字塔，用于对象检测和实例细分。
* 采用 adamw 优化算法对 COCO 模型进行训练：
  + 先对所有模型进行 100 次迭代预热，其中前 10 代进行线性预热。
  + 而后通过余弦学习速率将损失衰减到 0。

②分割任务：

* 将预训练的 ViT 模型作为 Mask2former 的主主干，采用 AdamW 优化器以多边形学习率的时间表进行训练，与 ADE20K 上的分割层一起进行微调。
* 对于任何分割任务，将预训练的轻量级 ViT 模型 ViT - Tiny 和 ViT - Small 作为 SAM 主干的图像编码器，并在 SA - 1B 数据集上对 EfficientSAM 译码模型进行 5 次微调。

## 模型性能的评估验证：

对于本项目而言，我们希望模型在目标检测与实例分割，目标分类识别方面有卓越性能，对此将进行评估与验证。

评估验证包括模型本身验证与实际应用验证两方面。

1. 图像分类：为了评估我们提出的技术在图像分类任务上的有效性，我们将所提出的SAMI思想应用于ViT模型，并在ImageNet-1K上比较它们与基线的性能。
2. 对象检测和实例分割：我们将SAMI预训练的ViT主干扩展到下游对象检测和实例分割任务，并将其与COCO数据集上的预训练基线进行比较，以评估其功效。我们采用预先训练的ViT主干，并将它们适应于Mask R-CNN主干中的简单特征金字塔，以构建检测器ViTDet。
3. Semantic Segmentation：我们进一步将训练的骨架扩展到语义切分任务中评估其有效性。具体而言，我们使用ViT模型作为Mask 2former框架中的主干，以ADE 20 K数据集为基准。
4. 利用EfficientSAMs进行语义分割：语义分割任务是一个可提示的分割过程，以基于任何形式的提示生成分割掩码，包括点集、粗略框或掩码、自由文本。我们遵循SAM，重点关注COCO/LVIS上基于点和基于框的提示分割。我们现在测试我们的模型在分割任何任务上的泛化能力，包括零样本单点有效掩码评估和零样本实例分割。我们将SAMI预训练的轻量级骨干作为SAM的图像编码器，以构建高效的SAM。然后，我们对SA-1B数据集上的EfficientSAM进行了微调，并报告了零样本单点有效掩码评估和零样本实例分割的性能。对于本项目，可以通过在测试图片里面选中要分割的商品等物品验证性能。

对于本项目而言，以上四项还可以与未经过优化实验的模型进行对比，进一步认证模型的有效性。

由于项目最终要求该模型能够有效且高效完成目标检测识别的任务，我们可以将该模型部署到机器人当中，通过观察对实地或仿真场景的处理结果，验证该模型的实际可行性。

# 位姿估计及位姿跟踪技术数据采集

## 基于SwinDRNet的深度补全网络

对于位姿估计网络来说，仅仅恢复可以看得见的部分的深度信息就足够进行特征匹配，参考SwinDRNet，**输入部分为rgb和深度图**，**输出得到经过修复的深度图。**

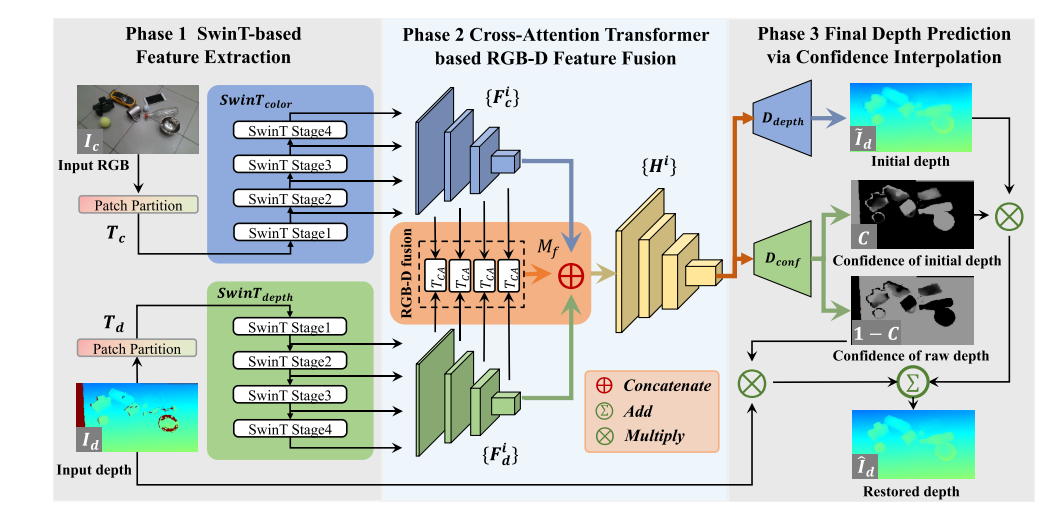
****

图7：模型整体架构

如图7所示，SwinDRNet的架构是一个双流融合编码器-解码器，可以进一步分为三个阶段：

1. 利用两个独立的SwinT主干分别从输入RGB图像和深度中提取分层特征;
2. 提出一个融合模块，其利用交叉注意变换器来联合收割机来自两个流的特征并生成融合的分层特征 ;
3. 提出了两个解码器模块，深度解码器模块Ddepth将融合的特征解码为原始深度，而置信度解码器模块Dconf输出预测的原始深度的置信度图，并且可以从输出通过使用置信度来计算最终恢复的深度。在输入深度的噪声和无效区域选择准确的深度预测，同时尽可能保持原始正确区域。

## 基于Foundationpose的位姿估计网络

Foundationpose具有良好的泛化性，并且已经集成到NVIDIA Isaac Manipulator中，在3090上Foundationpose初次位姿估计约1.3s，后续位姿跟踪可以达到30hz。后续可以先使用部分商品模型测试Foundationpose的零样本性能，若位姿估计性能无法达到要求，再使用备用方案SAM6DA微调。

## （补）基于SAM6D的位姿估计网络

补充6D位姿估计网络SAM-6D（3090上推理时间约4.37s）；

SAM-6D 通过两个步骤来实现零样本 6D 物体姿态估计，首先利用来自微调后的SAM网络（ISM）结合物体匹配分数实现任意物体的实例分割，接着通过位姿估计网络（PEM）的两阶段点集匹配过程获取物体的姿态。（注：截至2024/9/20在BOP榜单中关于未见过物体的位姿估计排名第十，前1-9的算法不开源或未公布训练代码）

**模型输入:rgb，depth，mask，3d-model，相机内参；**

**模型输出：物体pose；**

**网络结构可示意为：**

**实例分割提取——>基于点集匹配的粗姿态估计——>基于点集变换器的姿态优化——>输出pose。**

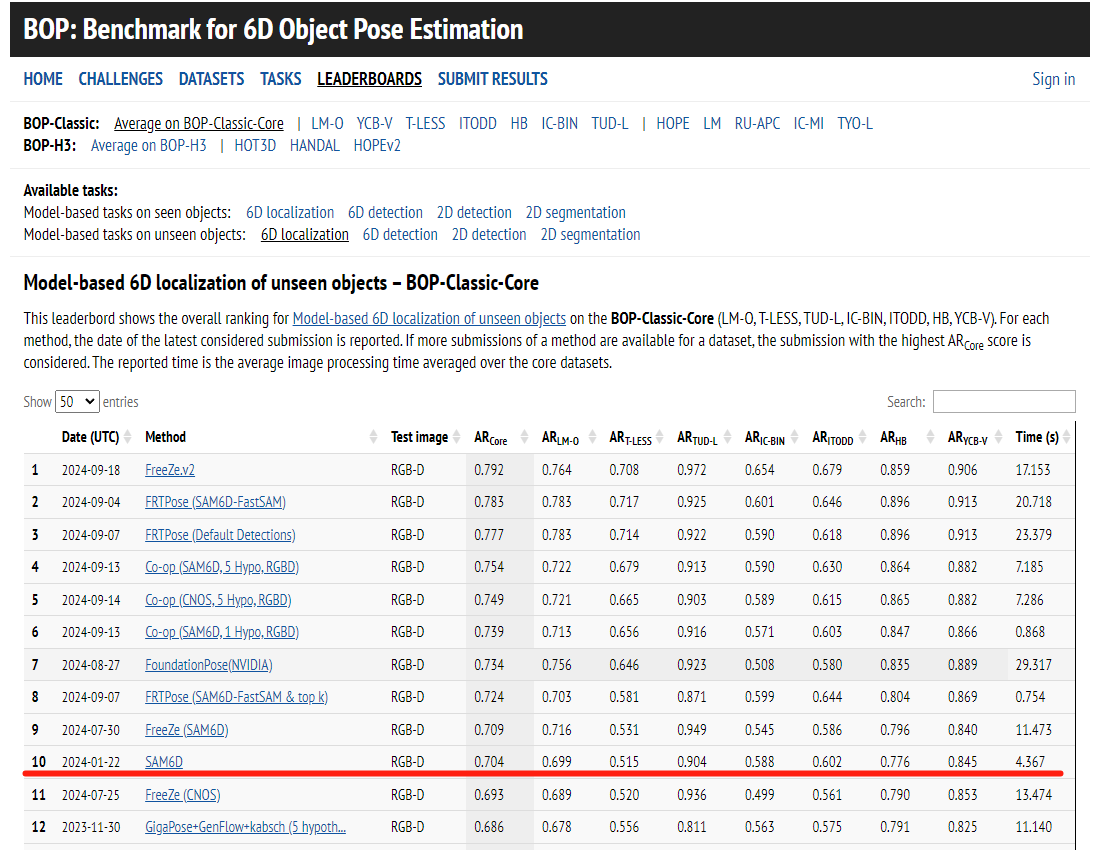


图8：Bop对于未见过物体位姿估计排名

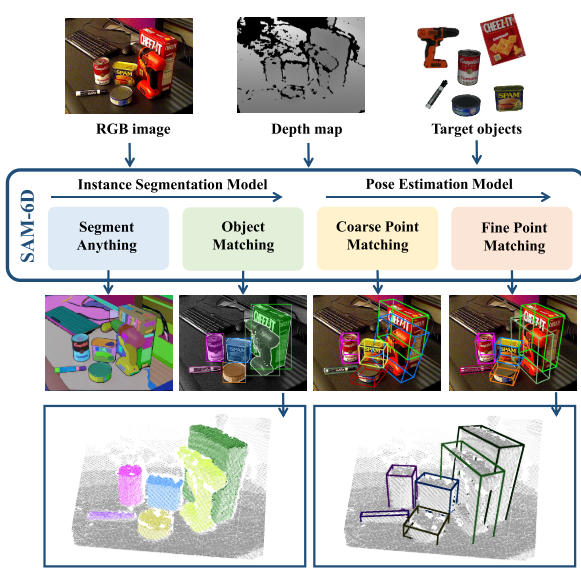


图9：模型总览示意图

SAM-6D包括一个实例分割模型（ISM）和一个位姿估计模型（PEM）。由于本项目的目标检测模型直接生成所需操作的商品的实力分割，基于此，可以跳过SAM-6D模型中这一模块，根据实例分割结果，直接进入位姿估计网络（PEM）。

1. 粗位姿估计网络

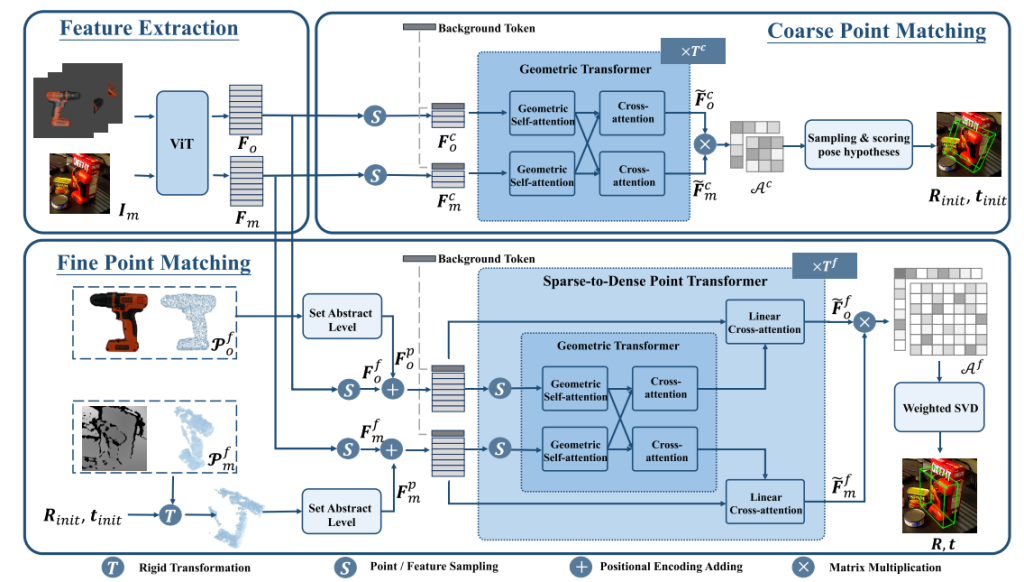


图10：位姿估计网络结构图

对于一个实例分割后的对象的物体的CAD模型的采样点集分别表示为和，其中和表示它们点的数量；同时，将这两个点集的特征表示为和，*C*表示特征通道数，PEM的目标是得到一个分配矩阵，表示从到之间的局部到局部的对应关系，由于遮挡或者分割不准确的情况，和只部分匹配。

为了解决两个点集非重叠问题，给输入图片的进行网格划分并配备Background Token，记为和，则可以基于特征相似性计算局部到局部的对应关系，首先计算注意力矩阵：

*，*

接着计算分配矩阵

，

其中，和分别表示沿着行和列进行softmax操作，是一个可控制的常数，一旦计算获得，则可以聚集所有配对点以及它们的匹配分数，最终利用加权SVD计算物体位姿。为每个粗姿态估计假设和计算姿态匹配得分：

在姿态假设中，选择具有最高姿态匹配分数的姿态假设作为输入到下一个细匹配模块中的初始姿态和。

1. 细位姿估计网络

细匹配模块用于建立密集对应关系并估计更精确的姿态R和t。为了建立更加精细的对应关系，我们选取一个更加稠密的采样点和。分别表示来自实例分割后的对象和物体的CAD模型。各自提取特征后，再分别注入包含位置编码的粗位姿估计和，利用针对点云的transformer注意力模块计算注意力矩阵，建立密集对应关系，算得姿态匹配得分。

## 6D位姿数据集采集

pose数据集需要RGB、深度、对象姿势、相机姿势、实例分割mask、模型信息等，采集难度较大，近期基于NVIDIA仿真合成数据集已经获得不错的影响力，例如模型DOPE仅仅使用由 Isaac Sim 的域随机化的合成数据。数据要求：

1. 数据集格式参照Bop，例如linemod，yvb-v，tless等；

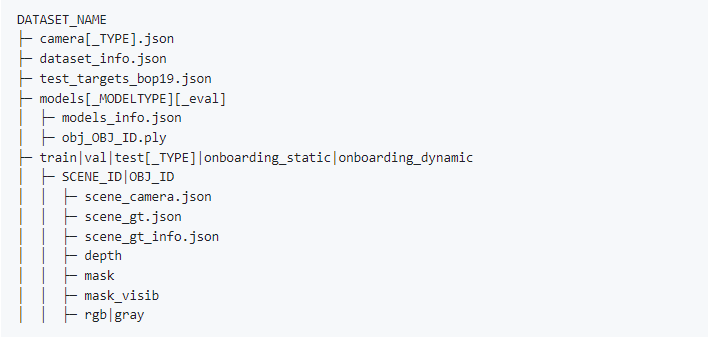


图11：Bop格式示例

1. 商品姿态主要包括放置状态和躺倒状态，以模拟我们的取货补货环境；
2. 包括商品部分遮挡，模拟物体在机械手上时相机获取不完整的情况；
3. 以市面上的公开数据集为参照，初期每种商品带有标注的图像至少1000张，并有对应depth，mask等。后续根据实验结果再增添数据集数量；

具体来说，数据采集按照标准6D位姿估计网络数据集，需要有以下字段：

*models[\_ MODELTYPE ]\_eval*  放置 3D 对象模型；

*train [\_ train type ]/ X*用来放置对象 X 的训练图像；

*val [\_ VALTYPE ]/ Y*用来放置场景 Y 的验证图像；

*test[\_ TESTTYPE ]/ Y*用来放置场景 Y 的测试图像；

*onboarding \_ static / obj \_ X \_ SIDE* 适用于无模型任务,对象 X 在上/下侧的静态入职图像；

*onboarding \_ dynamic / obj \_ X*仅用于无模型任务，对象 X 的动态 on board 图像；

*camera.json* 用来放置相机参数（仅用于传感器模拟；每个图像相机的参数在文件场景相机内。）；

其中，train, val和test文件夹中的图像被组织到子文件夹中:

*rgb /gray*：用来放置 彩色，灰色图像；

*depth*： 用来放置 深度图像（保存为16位无符号短格式）；

*mask* ：用来放置 对象轮廓的蒙版；

*mask\_ visib*（可选）：用来放置物体轮廓中可见部分的蒙版；

各个子区域的对应图像具有相同的ID，例如， rgb /000000.png 和 深度/ 000000 . png 是颜色和深度同一RGB-D帧的图像。mask的命名约定是IMID\_GTID.png，其中IMID是图像ID，GTID是Ground-truth真值注释的索引（存储在 json 中）。

**Camera parameters**

每组图像都附有文件*scene\_camera.json*，其中包含每幅图像的下列信息:

* *cam\_ K -*3x3内在摄像机矩阵K（按行保存）。
* *depth\_ scale*-将深度图像与此因子相乘，得到以mm为单位的深度。
* *cam\_ R\_ w2c* （可选）-3x3旋转矩阵R\_ w2c（按行保存）。
* *cam\_ t\_ w2c*（可选）-3x1位移向量t\_ w2c。
* *View \_ level*(可选)-视点细分级别,。

P\_w2i = K \* [R\_w2c, t\_w2c] 是相机矩阵，将世界坐标系中的 3D 点 p\_w = [x, y, z, 1]' 转换为图像坐标系中的 2D 点 p\_i = [u, v, 1]'：s \* p\_i = P\_w2i \* p\_w。**Ground-truth annotations**

真实物体Ground-truth姿态在文件 场景scene\_gt.json 中提供，其中包含每个注释的物体实例的以下信息:

* *obj\_ id* 表示对象ID。
* *cam\_ R\_ m2c* -3x3 表示旋转矩阵R\_ m2c（按行保存）。
* *cam\_ t\_ m2c* -3x1 表示平移向量t\_ m\_2c。

P\_m2i = K \* [R\_m2c, t\_m2c] 是相机矩阵，将模型坐标系中的 3D 点 p\_m = [x, y, z, 1]' 转换为图像坐标系中的 2D 点 p\_i = [u, v, 1]'：s \* p\_i = P\_m2i \* p\_m。

在*scene\_gt\_coco.json*下还需提供 COCO 格式的边界框和实例分割 。COCO格式见**1.1.1数据集采集**

**Meta information about the ground-truth poses**

文件*scene\_gt\_info.json*中提供了有关地面真实姿势的以下元信息 （使用BOPtoolkit 的scripts/calc\_gt\_info.py计算，其中 ITODD 数据集的 delta = 5mm，其他数据集的 delta = 15mm，为 BOP Challenge 2020 提供的所有真实感训练图像的 delta = 5mm）：

* *bbox\_obj* - 由 (x, y, width, height) 给出的物体轮廓的 2D 边界框，其中 (x, y) 是边界框的左上角。
* *bbox\_visib*——物体轮廓可见部分的二维边界框。
* *px\_count\_all*-物体轮廓中的像素数。
* *px\_count\_valid* - 物体轮廓中具有有效深度测量的像素数（即深度图像中的非零值）。
* *px\_count\_visib*- 物体轮廓可见部分的像素数。
* *visib\_fract* - 物体轮廓的可见部分 (= px\_count\_visib / px\_count \_all )。

**Acquisition of training images**

大多数数据集包括的训练图像是通过从各个视点捕获真实物体或通过渲染 3D 物体模型（使用BOPtoolkit的scripts/render\_train\_imgs.py）获得的。

渲染对象的视点是通过递归细分二十面体从视球中采样的，添加视点的细分级别保存在scene\_camera.json中作为 view\_level（对应于二十面体顶点的视点的 view\_level = 0，在第一个细分步骤中获得的视点的 view\_level = 1，等等）。为了减少视点数量，同时保持它们在球体表面上的“均匀”分布，可以仅考虑view\_level <= n 的视点，其中 n 是考虑的最高细分级别。

对于渲染，视场半径设置为所有测试图像中任何带注释的对象实例的最近出现的距离。该距离是从相机中心到模型坐标系原点计算的。

**3D object models**

3D 对象模型以 PLY (ascii) 格式提供。所有模型都包含顶点法线。大多数模型还包括顶点颜色或顶点纹理坐标，纹理保存为单独的图像。顶点法线是使用 [MeshLab](http://meshlab.sourceforge.net/)计算的，作为与顶点相关的面法线的角度加权总和。

每个包含对象模型的文件夹都包含文件models\_info.json，其中包含每个对象模型的 3D 边界框和直径。直径计算为任何一对模型顶点之间的最大距离。

**Coordinate systems**

所有坐标系（模型、相机、世界）都是使用右手坐标系。在模型坐标系中，Z 轴向上（当物体“自然直立”时），原点与物体模型的3D边界框的中心重合。相机坐标系与[OpenCV](http://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html)中的一样，相机沿Z轴观察。

**Units**

* 深度图像：参见各个数据集中的文件camera.json/scene\_camera.json。
* 3D 物体模型：1 毫米
* 平移向量：1 毫米

## 实验方案

1. 实验目的

通过实验来评估我们提出的SAM-6D的性能。

1. 实验准备
2. Datasets 数据集：

在BOP基准的七大数据集( LM-O, T-LESS, TUD-L, IC-BIN, ITODD, HB, YCB-V )中评估了SAM-6D。PEM（姿态估计模型）在MegaPose提供的数据集上训练学习。后续使用本项目制作的数据集进行实验，下文实验数据均使用BOP数据集。

1. Implementation Details 实现细节：

对于ISM（实例分割），遵循BOP基准，使用目标识别网络获得的实例分割和相机获取的rgb图片，使用DINOv2提取类和patch embeding。

对于PEM（姿态估计模型），设置Nmc=Noc=196和Nmf=Nof=2048，使用InfoNCE Loss损失函数来监督PEM的学习。并且使用ADAM训练PEM，以提高PEM的训练效率。

1. Evaluation Metries 评价指标：

姿态估计（PEM）：可见表面差异VSD、最大对称感知表面距离MSSD、最大对称感知投影距离 MSPD这三个误差函数的平均召回率 。

1. 实验步骤
2. 未知对象的姿态估计：

与现有方法的比较：比较SAM-6D的PEM与具有代表性的方法（MegaPose、ZeroPose、GigaPose）

将没有时间密集型的基于视觉渲染的优化工具加持情况下的PEM 和各种用目标遮罩预测技术优处理过的方法在BOP基准的七大数据集上进行姿态估计，记录各方法对应的平均召回率（AR）；

将PEM 用各种目标遮罩预测技术分别处理（包括来自ISM的目标遮罩预测技术）后，分别在BOP基准的七大数据集上进行姿态估计，记录对应的平均召回率（AR）。最后根据所得数据列出图表。

消融实验的研究 ：使用基于SAM的由ISM生成的目标遮罩预测技术 对YCB-V数据集进行消融实验，目的是评估单个设计在PEM中的有效性（功效）。使用设备：一台使用GeForce RTX 3090 GPU的服务器。

1. 研究 Background Tokens 的有效性：YCB-V姿态估计模型中，分别使用 Background Tokens和使用迭代优化策略的最优传输两种方法在YCB-V数据集上进行推理，分别记录对应的平均召回率（AR）和 整个数据集的平均每张图像处理时间（Time(s)）, 最后根据得出的数据列出图表。
2. 研究Two Point Matching Stages 的有效性：

i）在Background Tokens的基础上，通过粗点匹配模块和细点匹配模块设计了具有两个阶段的点匹配的PEM模型。将这个具有粗点匹配模块和细点匹配模块的PEM模型在YCB-V数据集上进行推导处理，记录对应的平均召回率（AR）；

ii）从PEM中移除粗点匹配模块，将移除粗点匹配模块的PEM模型在YCB-V数据集上进行推导处理，记录对应的平均召回率（AR）；

iii）从PEM中移除细点匹配模块，将移除细点匹配模块的PEM模型在YCB-V数据集上进行推导处理，记录对应的平均召回率（AR）；

iv）最后根据得出的数据列出图表。

1. 研究Sparse-to-Dense Point Transformers 的有效性：

在精点匹配模块中设计稀疏到密集点变压器（SDPT），以管理密集点的交互。

i) 在SDPT中，使用Geometric Transformers学习稀疏点集之间的关系，使用196个点的稀疏点集对Geometric Transformers进行实验，记录该点集中的点数和对应的平均召回率（AR）；

ii）在SDPT中，使用LinearTransformers将稀疏点集扩展到密集点集，使用2048个点的密集点集对LinearTransformers进行实验，记录该点集中的点数和对应的平均召回率（AR）；

iii) 在SDPT中，使用稀疏到密集点变压器（SDPT）管理密集点的交互，开始时先使用196个点的稀疏点集对SDPT进行实验，然后再使用2048个点的密集点集对SDPT进行实验，记录点集的点数变化和对应的平均召回率（AR）。

iv）最后根据得出的数据列出图表。

1. 实验数据分析
2. 分析—与现有方法的比较：

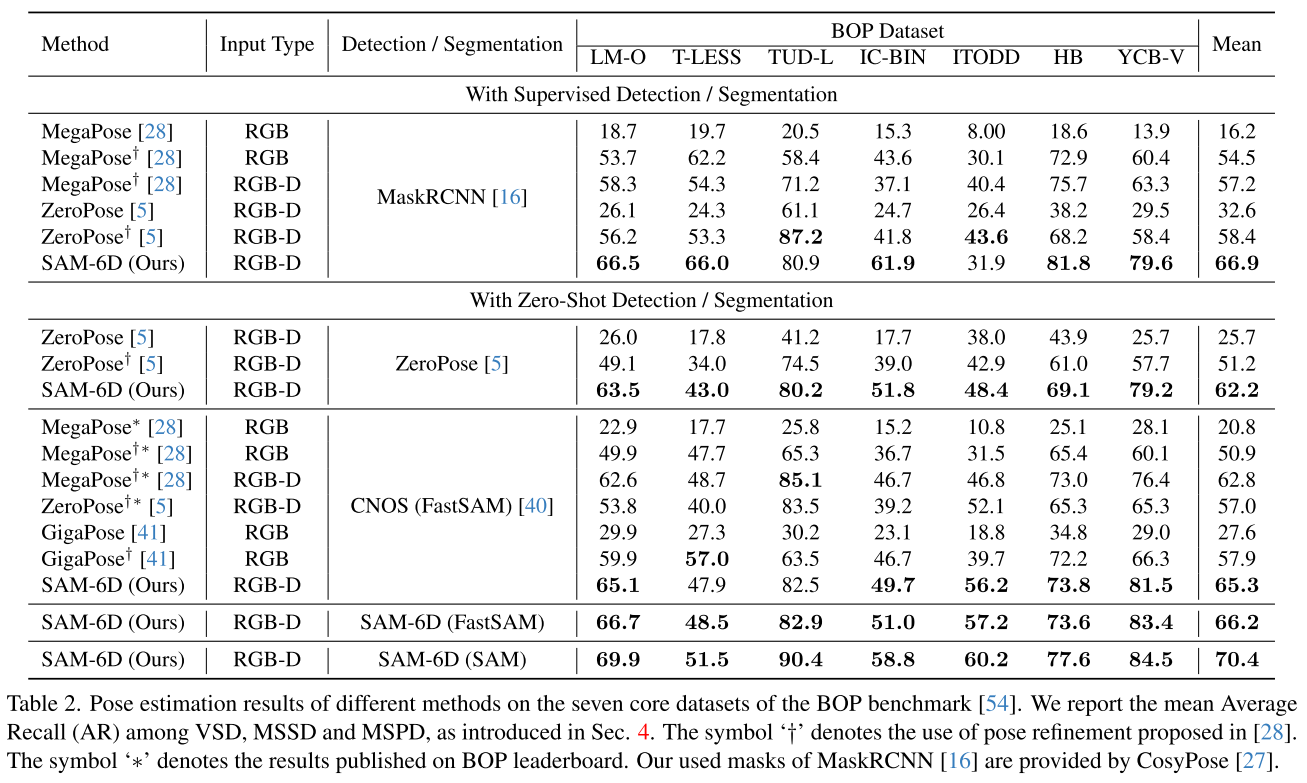


图12：与现有方法比较

结论：我们设计的PEM在没有用时间密集型的基于视觉渲染的优化、精炼模型或数据的工具或方法的情况下，性能优于其他使用各种目标遮罩预测技术优处理过的方法。将第六行与前五行的数据对比，SAM-6D在没有使用优化器的情况下Mean为66.9，而其他方法在用各种目标遮罩预测技术处理过后的Mean都不超过60.0。

结论：与使用其他目标遮罩预测技术相比，用目标遮罩预测技术能更显著提高PEM的性能。（进一步表明ISM的优越性）分析：将第六行、第九行、第十六行、第十七行和第十八行数据对比，可以发现：使用来自ISM的目标遮罩预测技术的Mean为70.4，超过其他四个Mean的值。

1. 消融实验的研究分析：

分析Background Tokens 的有效性：

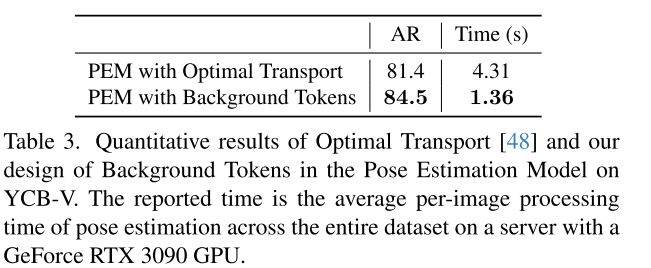


图13：Background Tokens 的有效性

结论：带有background tokens的PEM能实现与最优传输相当的效果，而且它推理速度更快。分析：在PEM模型中，解决部分到部分的点匹配问题共有两种方法：1、background tokens ；2、使用迭代优化策略的最优传输。由实验得出的图表，我们发现：带有background tokens的PEM和带有 最优运输 的PEM两者对应得出的AR数据相当，然而带有background tokens的PEM推理速度为1.36，带有 最优运输 的PEM的推理速度为4.31，即带有background tokens的PEM推理速度更快。同时也说明了background tokens对于PEM的有效性。

1. 分析Two Point Matching Stages 的有效性：

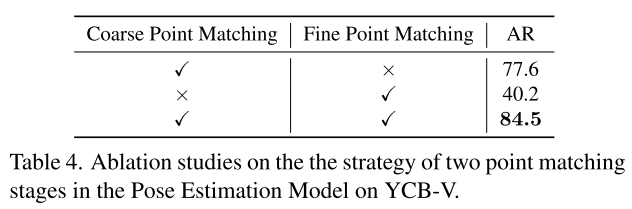


图14：Two Point Matching Stages 的有效性

结论：细点匹配模块和粗点匹配模块都对PEM有效。去除粗点匹配模块会显著降低PEM性能。分析：当PEM都具有细点匹配模块和粗点匹配模块时，对应平均召回率AR为84.5。当去除细点匹配模块时，对应的AR为77.6，可以验证细点匹配模块对于PEM的有效性；当去除粗点匹配模块时，对应的AR为40.2，与原来的数据84.5相比大大降低，可以验证粗点匹配模块的有效性，同时可以得出：去除粗点匹配模块会显著降低PEM性能。

1. 分析Sparse-to-Dense Point Transformers 的有效性：

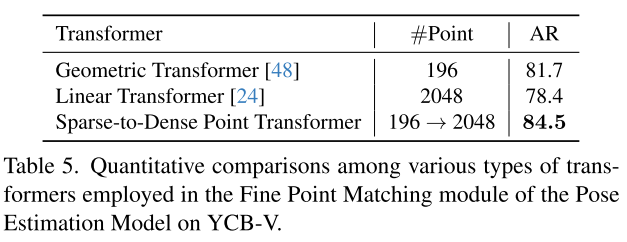


图15：Sparse-to-Dense Point Transformers

结论：Geometric Transformers与LinearTransformers与我们使用的SDPT相比，性能较差。SDPT对于PEM有效。分析：使用Geometric Transformers学习稀疏点集之间的关系后的PEM处理YCB-V数据集的平均召回率（AR）为81.7；使用Linear Transformers将稀疏点集扩展到密集点集后的PEM处理YCB-V数据集的平均召回率（AR）为78.4；而使用SDPT管理密集点的交互后的PEM处理数据集的平均召回率(AR)为84.5，超过前两者的AR。以上分析表明了Geometric Transformers与Linear Transformers与我们使用的SDPT相比，性能较差。同时也说明了SDPT对于PEM的有效性。

运行时的分析：

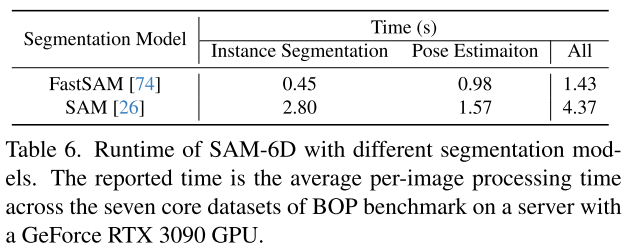


图16：在3090上不同网络下的运行时间

结论：使用SAM-6D处理数据的效率高，并且基于SAM的方法在姿态估计上用时比基于FastSAM的长。分析：SAM-6D基于FastSAM与SAM方法在姿态估计上用时分别为1.43、4.37。总体来说用时较短，避免了使用耗时的基于视觉渲染的优化工具。并且SAM-6D基于FastSAM的方法用时1.43，远少于基于SAM的方法用时4.37。

1. 实验结论

本实验探究SAM-6D框架的模型性能，该框架由实例分割模型（ISM）和姿态估计（PEM）组成，分两步完成任务。ISM利用SAM对所有包含潜在目标对象的候选区域进行分割，并根据语义、外观和几何尝试为每个目标对象匹配分数。然后，PEM通过粗点匹配和细点匹配两个阶段来解决部分到部分的点匹配问题，从而预测每个目标对象的姿态。SAM-6D的有效性在BOP基准中的七大数据集上得到了验证。SAM-6D的性能显著优于其他现有方法。

# 地图感知和路径规划规划

## 超市场景离线建图和建模

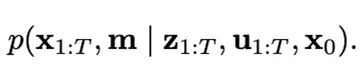
在系统的设计和实际运行中，无论是中央调度系统的路径规划，还是移动平台收到指令和路线后进行作业，都要基于超市或者其他应用场景的地图建模。一个准确且实时维护的地图能够保证整套系统的流畅稳定运作。SLAM(simultaneous localization and mapping, 同时定位和建图)算法技术是目前主流且适合当前项目场景的方法。

不同于目前市面上类似扫地机器人等的SLAM应用方式，本项目的建图以及SLAM的应用（包括系统部署后基于SLAM的路径规划及运动避障）可以事先对作业场景进行建图。以本次项目应用场景为例来说明，在系统设计和部署过程中，可以通过人工输入操作或遥控传感等并人工进行校准的方式，事先建立一张准确的超市场景地图，其中包含有准确的货架序列以及对应坐标，并储存在货物管理和中央调度系统以及边缘控制终端当中。无需像扫地机器人那样，首次运作需要对作业区域进行全区域巡视探测建图且无人工校准验证，无法保证地图精度和准确度。

其中关键在于，在建图和更新维护图及其节点信息的时候，如何通过传感器感知周围环境并提取相关特征，包括几何特征和深度信息等。根据调研分析，采用平面2D-SLAM技术。

SLAM是一种同时进行定位和地图构建的技术，可以在没有先验地图的情况下，通过使用传感器数据（如RGB和深度图像）来实时估计相机的位置，并构建环境的二维平面地图。

解决SLAM问题包括估计机器人轨迹和机器人在其中移动时的环境地图。由于传感器测量中的固有噪声，SLAM问题通常通过概率工具来描述。假设机器人在未知环境中沿着随机变量序列描述的轨迹x1:T={x1,...,xT}。在移动时，它会获取一系列里程计测量u1:T={u1,...,uT}和环境感知z1:T={z1,...,zT}。解决完整的SLAM问题包括估计机器人轨迹x1:T的后验概率和给定所有测量值加上初始位置x0的环境地图m：



初始位置x0定义了地图的位置，可以任意选择。为方便表示，在本文档的其余部分中，我们将省略x0。姿势x1:T和里程计u1:T通常在（特殊欧式群）SE(2)或SE(3)中表示为2D或3D变换，而地图可以用不同的方式表示。地图可以通过密集表示（如占用网格、表面地图）或原始传感器测量值参数化为一组空间定位的地标。特定地图表示的选择取决于所使用的传感器、环境特征和估计算法。在可以识别局部可区分特征的环境中，特别是在使用相机时，地标地图，通常是首选。相比之下，密集表示通常与距离传感器结合使用。与表示的类型无关，地图由测量值和获取这些测量值的位置定义。下图展示了3D和2D的三种典型密集地图表示：多级表面地图、点云和占用网格。

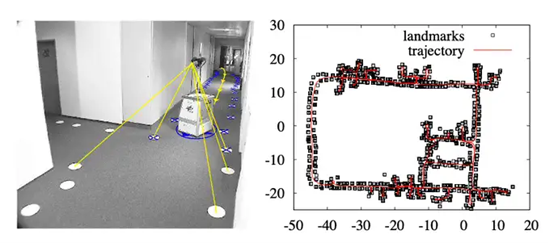


图11：典型的基于2D地标的地图

估计中给出的后验涉及在高维状态空间中操作。如果SLAM问题没有明确定义的结构，这将难以处理。这种结构源于某些常见的假设，即静态世界假设和马尔可夫假设。描述这种结构的一种方便方法是通过下图中描述的动态贝叶斯网络(DBN)。贝叶斯网络是一种图形模型，将随机过程描述为有向图。该图对过程中的每个随机变量都有一个节点，两个节点之间的有向边（或箭头）模拟了它们之间的条件依赖。

图片包含 日程表

描述已自动生成

图13：贝叶斯网络

在上图中，可以区分表示观察变量的蓝色/灰色节点（此处为z1:T和u1:T）和作为隐藏变量的白色节点。隐藏变量x1:T和m对机器人的轨迹和环境地图进行建模。DBN的连通性遵循以状态转换模型和观测模型为特征的循环模式。转移模型p(xt|xt−1,ut)由通向xt的两条边表示，并表示机器人在时间t在xt中的概率，假设在时间t-1它在xt中并且它获得了里程计测量ut。

观测模型p(zt|xt,mt)模拟了执行观测zt的概率，假设机器人位于地图中的位置xt。它由输入zt的箭头表示。外部观察zt仅取决于机器人的当前位置xt和（静态）地图m。将SLAM表示为DBN突出了其时间结构，因此这种形式非常适合描述可用于解决SLAM问题的过滤过程。

DBN的另一种表示是通过所谓的“基于图”或“基于网络”的SLAM问题公式，它突出了潜在的空间结构。在基于图的SLAM中，机器人的姿势由图中的节点建模，并标有它们在环境中的位置。由观察zt或里程计测量ut产生的姿势之间的空间约束被编码在节点之间的边缘中。更详细地说，基于图的SLAM算法从原始传感器测量值构建图。图中的每个节点代表一个机器人位置和在该位置获取的测量值。两个节点之间的边表示与两个机器人姿势相关的空间约束。约束在于两个姿势之间的相对变换的概率分布。这些转换要么是连续机器人位置之间的里程测量，要么是通过对齐在两个机器人位置处获得的观察结果来确定的。一旦构建了图，就会试图找到最能满足约束的机器人姿势的配置。因此，在基于图的SLAM中，问题在两个任务中解耦：从原始测量值构建图（图构造），确定给定图边缘的最可能的姿势配置（图优化）。图构造通常称为前端，它严重依赖于传感器，而第二部分称为后端，依赖于与传感器无关的数据的抽象表示。第V-A节描述了二维激光SLAM前端的简短示例。在本教程中，我们将描述一个易于实现但高效的基于图形的SLAM后端。

下图描绘了未校正的位姿图和相应的校正后的位姿图。

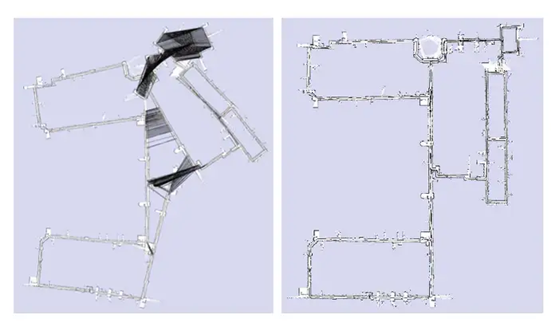


图14：未校正的位姿图和相应的校正后的位姿图

## 货架及货架上取货补货点位建模

在系统运行作业过程中，货架货物取放操作是最关键的一环。实现货架货物取放的立体操作关键也在于SLAM，但是不同于前文提到的对作业区域场景建图和路径规划过程中所涉及的SLAM技术，该环节的核心SLAM技术是对以货架为主要目标包括周边一定范围在内的区域进行立体建模建图仿真（路径规划过程中的SLAM是平面2D建模）。

3D-SLAM是一种同时进行定位和地图构建的技术，可以在没有先验地图的情况下，通过使用传感器数据（如RGB和深度图像）来实时估计相机的位置，并构建环境的三维地图。其中稠密SLAM是指在地图构建过程中，对环境的细节进行高精度的建模，包括对每个像素点的深度信息进行估计。

这里计划在项目中应用新型SLAM系统 SplaTAM ，它利用 3D高斯光滑函数作为地图的表示方法，实现了更快的渲染和优化速度，对地图的空间范围有明确的了解，并简化了地图的稠密化过程。

Spla TAM是一种用于稠密RGB-D SLAM的新型系统。它使用3D高斯散列表示法（3D高斯散列表示法是一种用于地图建模的方法，它使用高斯混合模型来表示三维空间中的数据。这种方法基于高斯函数，每个高斯函数代表了三维空间中的一个区域或体素，并包括位置、方向、大小等参数。通过将多个高斯函数组合在一起，以此来描述环境的结构，这种表示法可以用于建立稠密的环境地图，同时对环境的细节进行建模。通过这种方法，可以更准确地对环境进行建模和重建，尤其适用于需要高精度和细节的场景）实现密集的SLAM系统。

该方法通过在线优化明确体积表示，使用可微分的渲染实现快速优化和优化，并具有确定哪些区域以前被映射以及具有结构化的地图扩展功能（这种方法指的是一种通过在线的优化方式，对环境的体积表示进行建模。这种体积表示方法可以通过渲染技术进行快速优化，并且它具有可追踪哪些区域曾经被映射以及如何扩展地图的结构化功能。在这种方法中，通过渲染地图来进行实时的优化，以便更好地了解先前的映射情况，并且能够有机地扩展地图的结构，使其更加准确和结构化。这种方式能够使地图的构建更为高效，并且提供更多的可追溯性和结构化特性）。

同时实验结果表明，SplaTAM在相机姿态估计、地图构建和新视角合成方面取得了最先进的性能，超过了以往的方法。此外，SplaTAM还允许实时渲染高分辨率密集的3D地图,如下图所示，该方法适用于对本次项目应用场景。

图片包含 游戏机, 乐高, 桌子

描述已自动生成

图15：SplaTAM实时渲染高分辨率密集3D地图

SplaTAM相对于传统的视觉SLAM具有几个不同之处：

a) 地图表示方法：

SplaTAM采用3D高斯光滑函数作为地图的表示方法。传统的视觉SLAM通常使用稀疏或半稠密的特征地图，如特征点或关键帧。而SplaTAM的地图表示更加连续和光滑，使用高斯函数对地图进行参数化，这种表示可以更好地描述场景的连续性和光滑性。

b) 渲染和优化速度：

由于使用了高斯光滑函数作为地图表示方法，SplaTAM在渲染和优化地图时可能具有更快的速度。传统的SLAM系统可能需要在稀疏特征上进行优化和渲染，而SplaTAM使用高斯函数可能具有更高效的优化和渲染方法。

c) 地图空间范围的了解：

SplaTAM可能更容易对地图的空间范围有明确的了解。由于其地图表示方法的连续性和光滑性，可能更容易获取关于地图空间范围的信息，这对于一些应用场景可能非常重要。

d) 简化地图稠密化过程：

传统的SLAM系统可能需要在后期对地图进行稠密化处理，以获得更高精度的地图信息。SplaTAM可能会简化这个过程，因为它的地图表示方法可能已经相对连续且更接近实际场景，减少了对地图的后期处理需求。

SLAM系统 SplaTAM 使用基于3D高斯函数的辐射场（3D高斯函数的辐射场指的是利用高斯函数来描述三维空间内辐射或能量的分布情况）来进行Splat（渲染）、Track（跟踪）和Map（建图）操作。如下图所示。

截图里有图片

描述已自动生成

图16：3D高斯函数的辐射场

SplaTAM是一种用于稠密RGB-D SLAM的新型系统、它利用3D高斯光滑函数作为底层地图表示，实现了更快的渲染和优化、对地图空间范围的明确知识以及简化的地图密度化。同时SplaTAM在相机姿态估计、场景重建和新视角合成方面取得了最先进的结果。该方法通过将3D高斯光滑函数与SLAM相结合，不仅在SLAM和新视角合成领域树立了新的基准，还为进一步的探索和创新提供了一个稳健的框架：

地图表示方式： SplaTAM使用了3D高斯光滑函数作为底层地图的表示方法。这种方法相对于传统的离散地图表示，提供了更平滑、连续的地图表达，有利于更快的渲染和优化。

渲染和优化速度： 由于采用了3D高斯光滑函数，SplaTAM能够更快地进行地图渲染和优化，这意味着在SLAM系统中对地图的更新和重建可以更加高效。

空间范围的明确知识： 与离散化的地图不同，使用高斯光滑函数表示的地图能够提供对地图空间范围的更为清晰的认识，这对于姿态估计、场景重建和新视角合成等任务具有帮助。

场景合成和姿态估计的性能： SplaTAM在相机姿态估计、场景重建以及生成新视角方面取得了先进的结果，这表明它在处理视觉重建和场景合成方面具有优越性能。

核心方法

Gaussian Map Representation是一种用于表示场景的显式体素化地图表示方法。它使用一组3D高斯函数来表示场景的空间分布。每个高斯函数由其在三维空间中的位置、颜色和大小参数化。通过将这些高斯函数渲染成高保真度的颜色和深度图像，我们可以直接使用可微分渲染和基于梯度的优化来同时优化每个帧的相机姿态和场景的体素化地图。（每个高斯函数都代表了场景中的一个体素（三维像素）。它们描述了场景的局部信息，比如物体的位置、颜色和大小等。通过将这些高斯函数渲染成高保真度的颜色和深度图像，就可以直接使用可微分的渲染技术和基于梯度的优化方法，同时优化每个帧的相机姿态和场景的体素化地图）具体而言，Gaussian Map Representation的数学公式如下：

4.1、相机跟踪

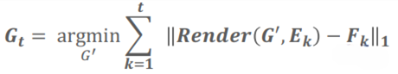


其中，E\_t表示当前帧的相机姿态，sil 表示相机姿态的损失函数，是权衡参数，𝑅𝑒𝑛𝑑𝑒𝑟表示渲染操作， G\_t-1表示上一帧的高斯地图，E' 表示上一帧的相机姿态。（我的理解中E'不应该是上一帧的相机姿态，应该是所要求的当前帧的相机姿态，要在这一系列的相机姿态里面找到使得误差最小的E'，这个E'就是E\_t）Render（G\_t-1，E'）表示对上一帧的高斯地图G\_t-1应用相机姿态变换E'之后的渲染结果。Ft是当前帧观察到的真实高斯地图。

（Sil>λ）意味着只有当Sil超过了某个设定的阈值λ时，才会考虑到这个误差，否则将不计算或对优化过程产生较小的影响。

对于整个公式，是一个优化问题。它的目标是找到当前帧的相机姿态(E\_t),使得通过上一帧地图和姿态转换后的预测地图（Render（Gt-1，E'）），与当前帧观察到的真实地图（F\_t）之间的误差最小化。

4.2、高斯密度化



其中，表示当前帧的高斯地图，𝐷𝑒𝑛𝑠𝑖𝑓𝑦表示高斯地图的密度化操作，𝑆𝑖𝑙表示密度化的损失函数。

4.3、地图更新

图片包含 徽标

描述已自动生成

其中，表示更新后的高斯地图，G' 表示上一帧的高斯地图， 表示第 k 帧的相机姿态， 表示第 k 帧的输入图像。

这个方程的目标是找到一个 G'，它在给定每个帧的相机姿态 Ek 下，用来渲染得到的图像与实际输入图像 Fk 之间的差异最小。通过比较渲染的图像和实际输入图像，我们可以计算误差的总和。通过不断地优化 G'，以使这些误差总和最小化，最终得到的 Gt 就是一个更准确、更符合实际的高斯地图，能更好地代表整个场景。

要找到一个 G'，使得对于每个帧的相机位姿 Ek，G' 渲染的图像与该帧实际观测到的图像之间的差异总和最小化。这样的话，这个更新后的 G' 在各个帧的位姿下都能更好地匹配实际观测到的图像，从而提高整体地图的准确性。

这些公式描述了SplaTAM方法中的关键步骤，包括相机跟踪、高斯密度化和地图更新。通过迭代优化这些步骤，我们可以同时估计相机姿态和优化场景的高斯地图，从而实现稠密的RGB-D SLAM。

5、相比于现有的地图表示方法，SplaTAM有何优势

快速渲染和优化：

高斯函数的渲染速度可以达到每秒400帧，比隐式方法更快速地进行可视化和优化。快速优化的关键在于对3D基元进行光栅化。

具有明确的空间范围的地图：

通过仅在过去观察到的场景部分中添加高斯函数，可以轻松控制现有地图的空间边界。（SplaTAM利用多个3D高斯函数来表示场景。这些高斯函数构成了地图的基本元素，每个函数代表了地图中的一个局部区域或体素。每个高斯函数在地图中代表了一个局部的空间范围，并且通过添加或移除高斯函数，你可以调整地图的空间范围和精度）

显式地图：

我们可以通过简单地添加更多的高斯函数来任意增加地图容量。

参数的直接梯度流动：

因为相机运动可以被视为保持相机静止并移动场景，所以我们还可以直接将梯度传递到相机参数中，从而实现快速优化。而基于神经网络的表示方法则无法做到这一点，因为梯度需要通过（可能是多个）非线性神经网络层传递。（当使用某些优化方法时，参数的直接梯度流动意味着可以通过相机位姿参数直接传递梯度，这种传递方式可以快速地进行优化。相机的位姿可以直接更新，而不需要通过复杂的非线性神经网络层传递梯度）

6、实验结果

相机姿态估计实验

使用四个数据集（ScanNet++、Replica、TUM-RGBD和Orig-ScanNet）进行相机姿态估计的实验评估。与多个基线方法（包括Point-SLAM、ORB-SLAM3等）进行比较。使用平均绝对轨迹误差（ATE RMSE）作为评估指标。

结果表明，SplaTAM在所有数据集上都表现出色，相对于其他基线方法具有更好的性能。

渲染质量实验

使用ScanNet++数据集进行渲染质量的实验评估。评估训练视图和新视图的渲染效果。使用PSNR、SSIM、LPIPS和深度L1损失等指标来评估渲染质量。

结果表明，SplaTAM在训练视图和新视图的渲染中都能提供高保真度的性能，相对于其他方法具有竞争力。

## 机械臂操作避障/硬件控制

经过前期调研，针对本项目的应用场景，这里提出建立一个由前端和后台以及硬件端三个部分组成的SLAM系统。前端可以允许操作人员修改作业区域场景地图，并可以查看作业区域实时状况，包括区域货架位置、作业平台实时位置坐标、异常障碍物位置坐标及大小等信息；后台与数据库联动，接受来自前端和硬件端的传感器信息并进行解算处理，并实时更新维护场景地图，来保证地图信息的准确性、实时性、有效性；硬件端主要是依靠作业平台其上搭载的传感器，其中主要包括激光雷达、单/双目摄像机等，检测并收集平台周遭特别是行驶区域前方的环境特征并发送至后台。

其中后台建立 2/3D复合SLAM系统。作业平台的路径规划和行驶避障基于平面2D-SLAM技术实现；在行驶到指定位置，并对商品进行抓取放置等立体作业时，则基于立体3D-SLAM技术实现。

这里针对机械臂对商品进行立体操作的关键要点做一个补充。由于该项目的应用场景是一种典型的标准化自动化的作业过程，经过假设分析，认为在机械臂作业过程中主要有三个要点步骤：1.承载平台准确定位行驶至指定坐标位置；2.机械臂及边缘硬件控制系统联动，准确识别定位目标商品在货架上的点位及其形状；3.边缘硬件控制系统驱动机械臂进行作业。

a) 平台准确定位行驶

平台在行驶过程中，首先基于平面2D-SLAM技术和内置IMU传感器进行路径规划，即粗避障/定位。行驶过程中，在转弯以及接近目的地等关键点位，结合平台摄像头和其他传感器对平台位置和行驶状态进行校准修正并调整，以实现较高的行驶定位精度，预测精度误差不超过5cm。另外，为了实现全流程自动化作业并创造有利条件，规定平台行驶路线需尽可能与货架走向平行，并最终保证平台到达目标坐标位置时，其位置与目标商品点位连线在水平面上与货架垂直。

b) 识别定位目标商品点位及其形状

如前文所提到的立体3D-SLAM技术以及相应的SplaTAM方法，对货架以及周边一定范围在内的区域进行立体建模建图，当中包含有货架上每个商品点位信息。该部分结合技术方案中关于6D姿态估计和机械臂抓取操作部分来实现。

c) 机械臂驱动控制

该环节基于立体3D-SLAM技术，并结合SplaTAM方法实现。由于项目应用场景中货架结构基本相同，且平台定位行驶保证机械臂作业时目标商品点位都在正前方，则机械臂夹爪或吸盘都可以水平靠近货架商品并操作。在系统设计构建过程中，可以根据现场实际情况，事前获取货架结构参数信息，并对每一层（比如三层货架有 高、中、低 三种情况）的商品设计编写一套控制程序，在作业过程中，边缘控制终端根据收到的目标商品点位坐标，调用对应的程序驱动机械臂进行作业。这样将极大的避免异常情况（如机械奇点、碰撞等）发生，并降低设计开发难度和周期。

# [常见物体的实时抓取估计技术流程](#_Toc25105)

## 基于CenterGrasp的抓取网络

CenterGrasp是一个结合了对象感知和整体抓取的新型框架，它通过在连续的潜在空间中对形状和有效抓取进行encoder来学习常规的一般对象。它由一个 RGB-D 图像编码器和SGDF解码器组成，前者利用最新的进展来检测对象并推断其姿态和潜在特征，后者用于预测场景中每个对象的形状和抓取情况。

在本项目中，物体姿态分为未分拣的堆叠躺姿态和需要从货架上拿取的放置姿态，为了获得稳定无序抓取，偏向于重心方向生成的抓取姿态可以适应更多的商品，符合本项目的实际要求。

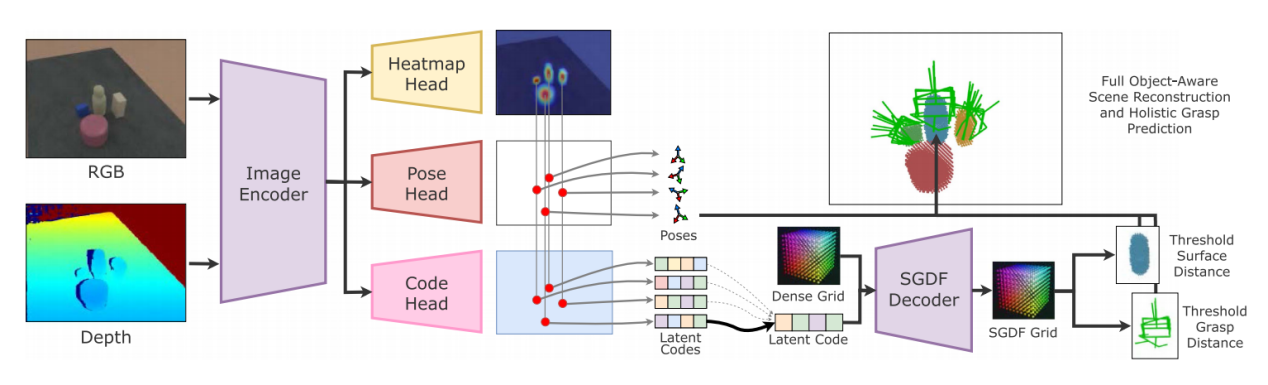


图 17：CenterGrasp网络架构图

**网络输入：rgb，depth**

**网络输出：抓取位姿**

**网络流程：**首先，将 RGB-D图像馈送到图像编码器中，该编码器输出对象热图、姿态图和潜在代码图。接下来，通过从预测的热图中提取峰值来确定图像中的对象位置。在这些位置，每个对象姿态和潜在代码都会相应地提取。在第二步中，SGDF解码器推断每个检测到的物体的形状和抓握。最后，使用对象姿态将形状和抓取预测从标准帧转换为相机帧。

1. 特征提取encoder

使用ResNet50拓扑生成RGB和深度图像的低分辨率特征表示。然后将这两种表示连接起来，并将它们馈送到ResNet18FPN主干。使用生成的特征金字塔作为三个专用头的输入：Heatmap-Head、Pose-Head和Code-Head。所有三个网络头都输出每像素预测。三个网络头的架构遵循PanopticFPN分段头。它将FPN特征作为输入，并将各级信息融合到单个输出中。每个特征级别被上采样到一个共同的维度，然后按元素求和。每个上采样阶段由3 × 3卷积、群范数、ReLU和2 ×双线性上采样组成。热图头将每个像素的对象概率预测为标量值o ∈ [0，1]。姿势头将给定像素处对象的6D姿势预测为12维向量，然后将其重塑为3 × 4矩阵并通过Procrustes操作投影到有效的齐次变换。

1. 形状和抓取估计decoder

形状和抓握距离函数（SGDF）解码器将3D坐标和潜在代码映射到形状距离和抓握距离。形状距离是一个标量值，表示采样坐标与对象曲面上最近点之间的带符号距离。抓取距离是最接近样本坐标的有效抓取的完整6D姿态，在与对象框架对齐并以样本坐标本身为中心的参考框架中表示。

得益于热力图的学习，网络可以较好的学习到物体中心的信息，使得无序抓取的抓取位姿生成更接近于物体重心，抓取的稳定性有所提升。

## [数据集采集](#_Toc26774)

Centergrasp仅在合成数据上进行训练，并实现了对真实的世界的零样本。训练均在GIGA的99，945个训练场景上进行训练。**基于本项目大多数抓取情况，可以在生成数据集时，设置相机位姿和物体放置随机生成姿态，使生成的数据集更符合本项目所需，提高抓取姿态生成能力**。下面说明GIGA训练场景数据集生成：

形状和抓取距离标签：为了为每个对象网格生成抓取候选项，首先从对象中采样1，000个表面点。对于每个点，将夹持器手指与点法线对齐。最后，对24个不同的手腕旋转进行均匀采样，即连续地围绕夹具的z轴旋转15度，每个对象总共有24，000个抓取候选对象。从这些候选中，我们计算地面真理把握标签检查碰撞和评估其antipodality。平均而言，数据集中每个物体大约有8，500次有效抓取。类似于NGDF，大的离散抓取集是有效抓取的连续流形的良好近似。每个对象使用mesh\_to\_sdf采样100，000个点及其各自的SDF值。对于每个点，我们从给定对象的ground-trurh抓取集合中找到最近的抓取，并计算相应的抓取距离标签。

图像标签：为了从预先计算的GIGA场景中生成合成图像观察（RGB和深度）以及姿势和实例分割标签，我们可以使用SAPIEN中基于光线跟踪的渲染器及其真实深度功能。所有的纹理、材质、灯光和桌子形状都是随机的。为了训练，我们可以从两个随机的相机姿势渲染每个场景，产生大约200，000个RGB-D图像和标签。对于姿态图和代码图标签，我们使用实例掩模分别用其地面真值姿态向量（12维）和地面真值代码向量（DI维）来标记对象的每个像素。

具体的来说，在GIGA仓库https://github.com/UT-Austin-RPL/GIGA存放着数据集生成代码，经过清洗，添加噪声等工作后，利用仓库https://github.com/robot-learning-freiburg/CenterGrasp的label、sgdf等生成脚本即可生成仿真数据集。

## 实验方案

**本实验流程是为了评估centergrasp模型性能，且为了与GIGA基线做对比，当前数据来自GIGA合成的仿真数据集**；

使用来自GIGA测试集的相同对象网格和随机场景，此外，通过在 YCB 对象集上进行测试来评估所有方法的分布外能力。生成两种不同类型的杂乱场景：堆积杂物，其中物体随机直立放置，以及堆叠杂物，其中物体随机掉落在桌子上并彼此重叠。从两个对象集和两种杂波类型的组合中，完整评估包括四种不同的环境。每个基线都针对每个环境中的 200 个随机场景进行评估，每个场景中的对象数是从预期值等于 4 的泊松分布中采样的。

**形状重建评估实验**：为了评估形状重建质量，使用真实网格及其在场景中的姿态来提取真实点云。然后，将它与 CenterGrasp 预测的点云以及从 GIGA 预测的网格中采样的点云进行比较。下图显示了两种方法的重建的定性比较。GIGA 在分布内评估（即使用 GIGA 对象）中产生了足够的结果，但在分布外设置中，其重建质量急剧下降。另一方面，CenterGrasp 在所有环境中都表现出良好的重建质量，包括实际评估。

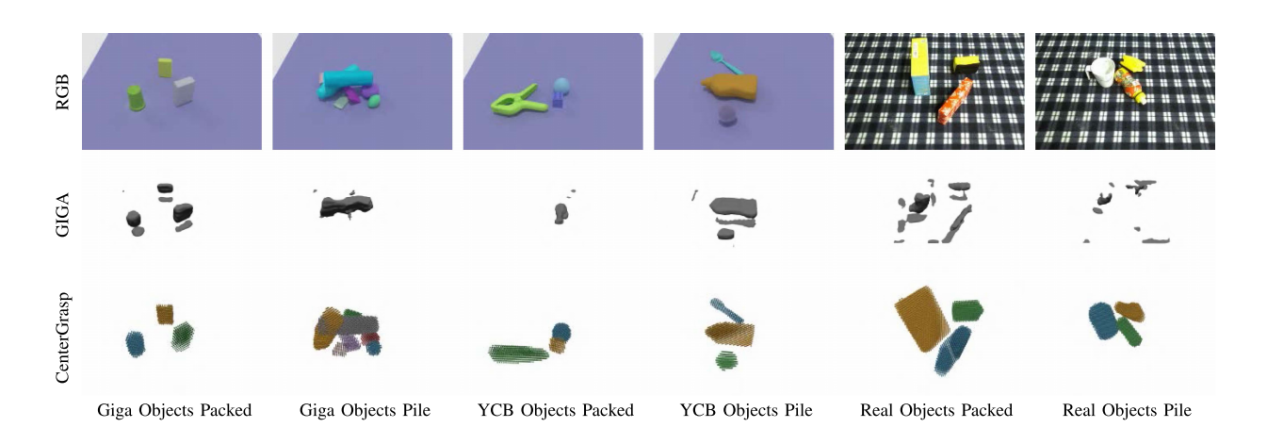
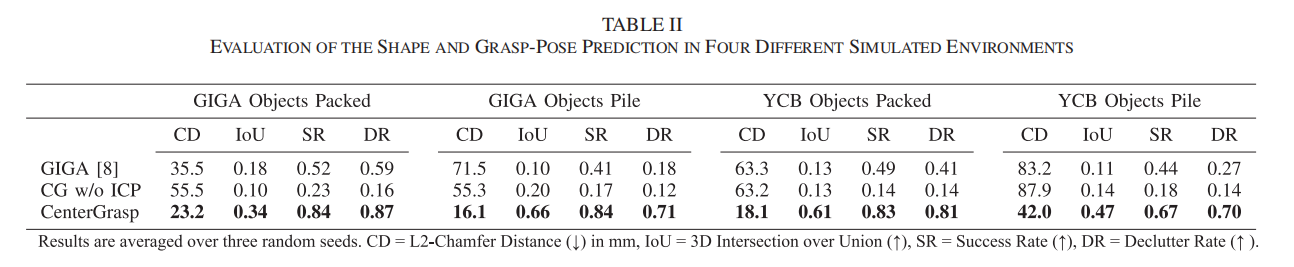


图18：CenterGrasp和GIGA对比

在下表中报告了评估的定量结果，其中包括在预测和地面实况之间计算的两个形状重建指标：L2 双向倒角距离 （CD） 和 3D 交并比 （IoU）。取平均，CenterGrasp 的 CD 提高了 38.5 毫米，IoU 分数提高了 0.38。该数值评估与图中的定性结果一致，突出了与最先进的技术相比，CenterGrasp 具有更好的泛化能力。这些结果证明了学习在对象级别而不是整个场景级别预测几何图形的优势。此外，对没有 ICP 的 CenterGrasp 的消融证明了姿势优化的重要性。结果表明，使用 YCB 对象评估时，没有 ICP 的 CenterGrasp 获得的结果与 GIGA 相似，而使用 GIGA 对象评估时获得的结果更差。

表1：定量评估结果

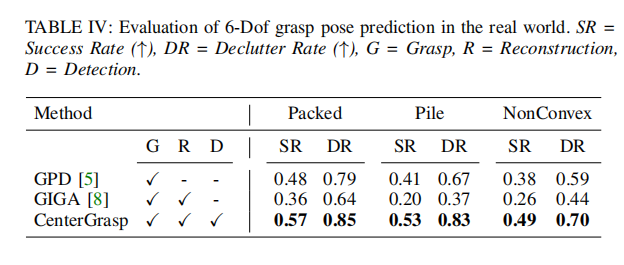


**抓取姿态预测评估实验**：对于每个场景，机器人都会执行预测的抓取，抬起物体，并将其移动到放置区域。如果对象成功移动到放置区域，则认为尝试成功。当场景中的所有对象都已清除或连续三次尝试失败后，将终止。在表2中，我们报告了两个对象抓取指标：成功率 （SR） 和整理率 （DR）。成功率定义为成功抓取次数除以尝试抓取的总次数，而整理率定义为成功抓取次数除以评估场景中的对象总数。后一个指标对于突出显示场景中仍然存在对象但模型无法预测任何抓取的情况非常重要。

在此实验中，与 GIGA 相比，CenterGrasp 的 SR 平均高出 33 个百分点，DR 高出 41 个百分点，这表明它能够从场景中清除更多的对象，同时需要更少的尝试。此外消融研究表明，在没有 ICP 的情况下使用 CenterGrasp 时，成功率显着降低，其效果明显差于 CenterGrasp 和 GIGA。与完整管道相比，没有 ICP 的 CenterGrasp 的 SR 和 DR 减少了 60 个百分点以上，凸显了精确物体姿势对整个管道成功的重要性。

**真实机器人实验：**该设置由配备平行钳口抓手和手腕式 ZED2 深度相机的 7-DoF Franka Emika Panda 手臂组成，并收集一组 20 个简单和一组 6 个非凸的家用物品。共评估三种设置：填充和堆积（类似于模拟实验）和非凸面，并构建了 30 个不同的场景，每个场景有三个对象，所有三个场景总共有 90 种可能的抓取方式。为了最大限度地减少差异，尽可能准确地复制所有方法的场景。对于每个场景捕获一个 RGB-D 图像并查询模型以预测抓取，然后执行预测的抓取并从场景中删除对象。如果同一对象的抓取尝试失败两次，则会手动从场景中删除该对象。对于每个基线和场景类型，表4 中报告了成功率 （SR） 和整理率 （DR）。与仿真结果类似， CenterGrasp 在所有评估中都表现出最佳性能，与 GIGA 相比，SR 平均高出 27 个百分点，DR 高出 20 个百分点。

表2：不同模型的抓取成功率



## 基于物体表面点云的吸取网络

1. **点云数据获取与采样**

使用深度摄像头获取物体的点云数据后，对点云数据进行滤波和去噪，确保数据的准确性和完整性。然后从点云中随机抽取一部分点，用于拟合平面。如果有多个物体，需分割点云**(可由1.1目标检测网络给出的mask分割点云)**，识别出每个独立物体的点云。

1. **平面拟合**

使用RANSAC算法来拟合平面，因为它对噪声和离群点具有鲁棒性，然后通过拟合的平面方程得到法向量。

1. **选择吸取点并验证**

将法向量作为吸盘末端的旋转姿态，以确保吸盘垂直于物体表面；根据拟合平面的中心点，确定吸取的表面位置。然后确认法向量方向和吸取点位置能够提供足够的吸附力和稳定性，并确保吸盘在抓取过程中不会与其他物体或环境发生碰撞。

## 实验方案

对于每个场景，机器人都会执行预测的吸取，抬起物体，并将其移动到放置区域。如果对象成功移动到放置区域，则认为尝试成功。当场景中的所有对象都已清除或连续三次尝试失败后，将终止。设置两个物体吸取指标：成功率和整理率。成功率定义为成功吸取次数除以尝试吸取的总次数，而整理率定义为成功吸取次数除以场景中的物体总数。后一个指标对于突出显示场景中仍然存在对象但无法预测任何吸取的情况非常重要。

对两组情况进行实验：物体正常放置和物体堆叠放置（类似于模拟实验），并构建30 个不同的场景，每个场景有三个物体，所有三个场景总共有 90 种可能的吸取方式。对于每个场景利用逆运动学计算机械臂的移动路径，将吸盘移动到目标位置，控制机械臂执行吸取动作，监测吸附情况。记录成功率、整理率、吸附力、时间效率等指标。分析失败案例，调整算法参数。并根据实验数据，优化吸取点选择策略，调整法向量计算和吸盘姿态，以提升抓取效果。

# [仿真验证](#_Toc21494)

## 基于ros的仿真平台验证

## 实验方案

ROS仿真平台验证实验

ROS（Robot Operating System）仿真平台验证实验是一种利用ROS框架进行机器人系统测试与验证的高效方法。ROS不仅仅是一个传统的操作系统，它是一个功能强大的开源机器人操作系统框架，提供了开发、测试、运行机器人应用程序所需的完整工具链与库支持。得益于其模块化设计，ROS支持各种机器人技术的集成与应用，使得研究人员和工程师能够轻松开发出复杂的机器人系统。ROS的广泛适用性、跨平台兼容性和灵活性为机器人开发提供了坚实基础，并在多个领域中展现了巨大的应用潜力。

强大的跨平台兼容性：操作系统与硬件支持

ROS的产业系统具有高度的跨平台兼容性，这使得它能够在不同操作系统上运行，从而满足多样化的开发需求。无论是Linux、Windows 10（通过WSL）还是macOS，开发者都可以根据项目需求选择最适合的操作系统进行开发和测试。

1. 操作系统兼容性

• Linux环境的主导地位：ROS的主要开发环境是Linux，特别是Ubuntu发行版，得益于其开源、稳定和高效的特点，Linux已经成为机器人领域的首选操作系统。在Linux上，开发者能够利用丰富的开源工具和社区资源，快速构建复杂的机器人应用。

• Windows 10与WSL的支持：为适应更多开发者的需求，ROS也支持Windows系统，尤其是通过WSL（Windows Subsystem for Linux）提供了近乎原生的Linux体验。这使得那些习惯微软开发生态的开发者能够轻松进入ROS的世界，无需改变其已有的开发工具链。

• macOS的兼容性：对于开发者而言，macOS的兼容性进一步扩展了ROS的应用范围。使用macOS的开发者可以轻松切换到ROS开发环境，尤其适合那些需要在Apple硬件平台上进行开发和测试的场景。

2. 硬件平台兼容性

ROS不仅支持多种操作系统，还能与广泛的硬件平台兼容，从小型机器人到大型工业设备，开发者都能够利用ROS进行开发。其生态系统支持多种传感器和执行器，包括激光雷达、深度相机、IMU等，为无人驾驶、工业自动化和家庭机器人等多种应用场景提供了硬件支持。这种高度兼容的设计让ROS成为工业和消费级机器人开发的理想选择。

仿真平台的多样性与应用广泛性

ROS仿真平台的广泛适用性是其在机器人开发中备受推崇的原因之一。ROS支持多个高性能仿真工具，如Gazebo和RViz，帮助开发者在虚拟环境中精确模拟机器人在各种复杂场景中的行为表现，包括导航、避障、环境感知等。以下是ROS仿真平台的主要优势：

• 虚拟实验室：通过ROS仿真平台，开发者可以在没有实际硬件的情况下，完成大规模的实验和验证。这尤其适用于那些涉及昂贵硬件的项目，如高精度传感器或复杂机械臂。开发者可以通过仿真工具，验证机器人系统在各种极端环境下的表现，例如在充满障碍物的路径中导航或在未知环境中进行探索。

• 快速原型设计与硬件替换：ROS的模块化架构允许开发者轻松添加或替换硬件组件，而无需对软件系统进行大幅修改。这种灵活性大大缩短了开发周期，加快了产品的原型设计和迭代过程。例如，开发者可以在仿真环境中快速测试新的传感器集成，确定其可行性后再投入实际硬件开发。

• 高效的算法验证：在仿真环境中，开发者可以测试并优化复杂的算法，如路径规划、SLAM（同步定位与地图构建）和物体识别。通过虚拟测试，开发者可以大幅降低硬件损坏风险，同时确保算法在不同场景下的鲁棒性和稳定性。

ROS仿真平台在产业系统中的应用

ROS仿真平台已经在多个产业领域展现出巨大的潜力，尤其是在工业自动化、无人驾驶、医疗机器人和服务机器人等领域。

• 工业自动化：在智能工厂中，工业机器人通常需要与多个传感器、执行器和控制系统协同工作。ROS提供了标准化的消息传递机制和模块化的架构，使得开发者能够快速集成这些设备，简化了复杂的系统集成过程。例如，利用ROS仿真平台，开发者可以在虚拟环境中测试流水线的控制算法，优化机器人臂的工作流程，从而减少停机时间并提高生产效率。

• 无人驾驶：在无人驾驶技术中，ROS仿真平台同样扮演着关键角色。开发者可以通过ROS仿真环境，验证车辆的路径规划、避障策略和决策算法。通过模拟不同的交通场景，开发者能够在真实车辆上进行测试之前，优化算法的性能并降低成本。

• 服务机器人：在家用和商业环境中，服务机器人越来越受到关注。借助ROS仿真平台，开发者可以模拟机器人在家庭或办公室中的导航和物体识别能力，验证其在不同环境下的行为表现。这种高效的仿真测试方式加速了机器人产品的上市速度。

高成本效益与安全性

传统机器人研发中，硬件设备成本往往非常高，尤其是在涉及昂贵的传感器和复杂机械结构时。而使用ROS仿真平台，开发团队可以在无需昂贵硬件的情况下进行早期开发和验证，大大降低了前期的投入成本。开发者仅需基础的计算资源，就能在仿真环境中开展复杂的测试与研究，从而实现高效、低成本的开发过程。此外，ROS仿真平台提供了一个安全的试验场景，特别是在处理危险或极端环境下任务时，如救援机器人或灾难场景中的作业机器人。通过在仿真环境中测试和优化算法，开发者可以规避物理损坏、设备损失和人员伤害的风险，同时确保系统的安全性和鲁棒性。

快速迭代与复杂场景模拟

ROS仿真平台的另一个显著优势在于其支持快速迭代与复杂场景模拟。开发者可以在仿真环境中快速验证不同设计思路，通过调整参数和配置，迅速测试新的想法与解决方案。借助ROS仿真平台的灵活性，开发者能够创建高度定制化的虚拟环境，从多机器人协作场景到复杂的城市交通模拟，甚至室内服务机器人的导航系统。这种高度拟真的测试环境不仅有助于开发者验证新技术的有效性，还能通过逐步优化算法，使得最终系统更加健壮和高效。每次仿真迭代都提供了有价值的反馈，帮助开发者不断改进系统，直至达到最佳解决方案。

全球开发者社区与开源生态

ROS的开源特性及其活跃的全球开发者社区，极大推动了其在各个领域的持续创新与发展。ROS生态系统中拥有大量开源项目和预构建的功能包，开发者可以直接利用这些资源，加速开发进程。全球范围内的开发者社区提供了丰富的文档、技术支持和合作机会，无论是初学者还是资深工程师，都能从中受益。此外，ROS的开源生态还鼓励开发者共享他们的经验和成果，通过合作共同推动机器人技术的进步。

1. 为什么选择ROS仿真平台验证实验？

成本效益

在机器人开发的早期阶段，仿真平台能够降低对昂贵硬件的依赖，显著节省成本。开发者可以通过仿真环境快速验证系统设计的可行性，为后期的硬件实现奠定基础。

安全性

仿真平台为危险任务和极端场景的机器人开发提供了安全测试环境，能够降低真实环境中的事故风险。通过在仿真环境中测试故障应对策略，开发者能够及时发现安全隐患并进行改进。

快速迭代

仿真环境支持快速设计调整和参数优化，使开发者能够高效完成多个迭代周期。与实际硬件相比，仿真平台为数据驱动的算法优化提供了灵活性和速度。

复杂环境模拟

ROS仿真平台能够模拟多样化的复杂场景，帮助开发者在虚拟环境中进行系统全面测试，确保机器人能够应对不同应用领域中的复杂任务。

2. 如何使用ROS仿真平台进行验证实验？

ROS仿真平台为机器人系统的开发和验证提供了一个灵活且强大的工具，能够帮助开发者在无需真实硬件的情况下进行大规模测试与验证。通过使用仿真工具，开发者可以测试算法、评估系统性能并进行多次迭代，优化系统设计。以下是使用ROS仿真平台进行验证实验的详细步骤：

1. 环境搭建

要进行ROS仿真实验，首先需要搭建合适的开发和仿真环境。以下是环境搭建的基本步骤：

• 安装ROS：在Ubuntu操作系统上安装合适版本的ROS（如Noetic），ROS是目前机器人开发的主流平台。可以通过使用ROS的包管理器apt-get快速安装所需的软件包。

• 安装Gazebo与RViz：Gazebo是ROS生态系统中的3D动态模拟器，用于创建物理仿真环境；RViz是一个可视化工具，用于查看和分析机器人运行状态。这两个工具是ROS仿真平台的核心组成部分，允许开发者在虚拟环境中进行交互式操作。

• 配置开发环境：在安装完毕后，将ROS环境添加到用户的.bashrc文件中，以便每次启动终端时都能自动加载ROS环境配置。使用source命令加载环境，使得开发者能够在终端中直接调用ROS相关工具和命令。

2. 创建仿真模型

在仿真中，机器人模型的创建是至关重要的。通过定义物理特性和仿真环境，开发者可以构建出符合需求的实验场景：

• 创建机器人模型：使用URDF（统一机器人描述格式）或SDF（仿真描述格式）来描述机器人的结构、传感器、执行器和物理特性。URDF通常用于定义机器人的形状和运动学，而SDF则用于更加详细的物理仿真，支持更复杂的物理属性。

• 设计仿真环境：在Gazebo中创建仿真场景，包括地形、障碍物和其他重要的环境元素，以模拟机器人在现实世界中的运行环境。开发者可以根据实验需求设计复杂的场景，如室内导航、工业自动化车间等。

• 导入模型：将定义好的URDF/SDF模型导入到Gazebo中，通过launch文件或Gazebo图形界面启动仿真。启动文件简化了模型加载和场景设置，确保系统的快速初始化。

3. 开发与集成算法

仿真平台的核心目的是测试和优化机器人控制算法，ROS提供了丰富的工具用于算法开发和集成：

• 编写控制代码：使用C++或Python编写控制机器人行为的程序，通过ROS的节点架构将各个算法模块封装为独立的ROS节点。每个节点负责执行特定的任务，比如传感器数据处理、路径规划、运动控制等。

• 配置ROS节点与通信机制：ROS节点之间通过话题（Topics）进行通信，节点能够订阅或发布传感器数据和控制指令。对于需要请求-响应模式的任务，可以使用ROS的服务（Service）机制；对于长时间运行的任务或带有反馈机制的任务，可以使用动作（Action）机制。

• 仿真参数配置：设置仿真的物理参数，如摩擦系数、重力加速度等，以确保仿真环境的行为与真实世界相似。根据不同的实验场景，调节传感器的参数或运动控制算法的配置。

4. 运行仿真实验

完成模型创建和算法开发后，开发者可以运行仿真实验，验证系统性能：

• 启动仿真环境：通过启动文件或命令行启动Gazebo，并加载机器人模型和仿真世界。ROS的roslaunch工具允许开发者在启动时自动初始化多个节点，简化启动流程。

• 监控实验进程：使用RViz可视化机器人状态，如传感器数据、路径规划和运动轨迹，观察机器人的行为是否符合预期。RViz不仅提供实时的可视化效果，还可以通过插件显示详细的传感器信息和仿真状态。

• 数据记录与日志分析：ROS提供了日志记录功能，能够记录机器人在仿真中的数据，以便后续分析和调试。开发者可以通过rosbag记录实验数据，随后在实验结束后进行详细分析。

5. 结果评估与优化

验证实验的最后一个关键环节是分析仿真结果并优化系统，以提升性能：

• 评估算法性能：分析实验结果，评估控制算法、路径规划算法等在仿真环境中的表现。衡量标准可以包括任务完成时间、能耗、路径长度、误差等。

• 识别问题与瓶颈：通过日志和实验数据，开发者可以发现系统中的问题，可能是由于算法中的逻辑错误或仿真环境中的配置不当。通过调试，识别性能瓶颈，并在下一次实验中进行改进。

• 迭代优化：根据评估结果，对算法进行调整或重新配置仿真环境的参数，然后重新运行实验。通过多次迭代，逐步优化系统设计，直到获得最佳性能。仿真平台提供了灵活的调试能力，使得快速迭代成为可能。

NVIDIA Isaac Sim

随着机器人技术的迅猛发展，对高效、精确且可扩展的仿真工具的需求日益增长。然而，传统的机器人开发和测试方法面临着高昂的硬件成本、安全隐患以及复杂场景难以重现等诸多挑战。这些因素不仅增加了研发时间和经济成本，还限制了研究人员在真实环境中进行充分测试和验证的能力。

为了解决这些挑战，NVIDIA Isaac Sim 应运而生，成为一款专为机器人仿真设计的强大工具。通过提供高保真的虚拟环境，Isaac Sim 显著降低了研发成本并提高了开发效率。它利用 NVIDIA 的 PhysX 引擎和 RTX 技术，创建出极其逼真的物理模拟和视觉渲染效果，使研究人员能够在完全受控的环境中探索更多可能性，从而加速创新步伐并确保算法和控制策略的可靠性。

对于希望降低研发成本的企业，Isaac Sim 提供了减少硬件投入和测试时间的有效解决方案。对于渴望探索前沿科学发现的研究机构，它不仅提供了前所未有的便利性和灵活性，还加速了创新，使研究人员能够快速验证和优化复杂的机器人算法。NVIDIA Isaac Sim 以其卓越的仿真精度，正在重新定义机器人研发的标准。

什么是NVIDIA Isaac Sim？

NVIDIA Isaac Sim 是一款基于 NVIDIA Omniverse 平台的高级机器人仿真工具，利用 PhysX 引擎和 RTX 技术提供了极其逼真的物理模拟与视觉渲染效果。它支持多种传感器模拟和与 ROS 的无缝集成，适用于物流、制造和自动驾驶等领域，帮助开发者在一个高度仿真的虚拟环境中高效地设计、测试和验证机器人算法。

NVIDIA Isaac Sim的特点

NVIDIA Isaac Sim 集成了物理引擎、传感器模拟、机器学习框架等多种先进技术，旨在简化从设计到部署整个机器人系统的过程。通过高度逼真的虚拟环境，Isaac Sim 支持开发者在安全、可控且成本效益高的环境中测试和验证复杂的机器人算法及控制策略。以下是 Isaac Sim 的主要特点：

高保真度物理仿真

• PhysX引擎：利用NVIDIA PhysX引擎的强大计算能力，Isaac Sim创建了极其接近现实世界的物理交互模型，涵盖重力、碰撞、摩擦等重要属性，确保仿真结果真实可靠。

• RTX渲染：结合 NVIDIA RTX 技术，Isaac Sim 生成高质量图像，提供逼真的光照和视觉效果。这不仅增强了传感器数据生成的真实性，还尤为重要于传感器数据的生成和验证，使虚拟环境与现实世界几乎无异。通过 RTX 渲染，开发者能够在高度仿真的环境中进行更精确的测试和验证，确保算法和控制策略在真实世界中的可靠性和有效性。

全面的传感器支持

• 多类型传感器支持：Isaac Sim支持RGB摄像头、深度摄像头（如RGB-D相机）、激光雷达（LiDAR）、惯性测量单元（IMU）等多种传感器的模拟，为训练计算机视觉算法、目标检测和路径规划提供必要的数据。

• 合成数据生成：内置的Replicator模块使用户能够快速生成大规模标注数据集，极大促进AI模型的研发，并支持领域随机化和数据增强，确保数据的多样性和真实性。

无缝集成ROS

• ROS兼容性：Isaac Sim 与 ROS（Robot Operating System）无缝集成，支持 ROS 2，允许现有项目直接迁移至 Isaac Sim 进行开发或测试。这种兼容性简化了从现有项目到仿真环境的迁移过程，使开发者能够轻松地在高度逼真的虚拟环境中进行工作。丰富的 ROS 接口和工具方便用户进行实时通信和数据交换，确保仿真环境中的测试和验证结果与真实世界高度一致，从而提高算法和控制策略的可靠性和有效性。

• 硬件在环测试：通过ROS桥接，支持硬件在环（HIL）测试，使开发者能够在仿真环境中验证实际硬件的行为，缩短开发周期并降低风险。

强化学习支持

• Isaac Gym：结合Isaac Gym组件，提供理想的实验平台，允许用户在高度逼真的环境中训练智能代理，优化其行为策略，促进复杂环境中的RL算法测试与改进。

成本效益

• 降低硬件成本：使用仿真环境显著降低开发和测试机器人的成本，避免昂贵的硬件需求。

• 节省时间和经济成本：在仿真环境中快速迭代算法，省去对真实机器人的繁琐调试，节省大量时间和成本。

安全性与可控性

• 安全性：在虚拟环境中测试机器人，避免潜在危险，保护人员和设备安全。

• 可重复性：仿真实验易于复制，确保结果的一致性，提供更可靠的验证基础。

• 可控性：用户能够控制合成环境的参数，测试算法在不同条件下的鲁棒性。

数据多样性与标定

• 数据多样性：通过合成数据生成多样的场景和环境条件，帮助训练和评估算法，尤其是那些难以获取的数据。

• 数据标定与真值：合成数据集可精确标定真实场景中的信息，提升算法性能。

挑战与局限

• 理想化假设：仿真环境常需引入理想化假设，可能导致真实世界性能与仿真结果不完全一致。

• 有限精确性：现代仿真工具受限于计算资源和物理模型，无法完美模拟所有机器人行为。

• 缺乏真实噪声：合成数据通常不具备真实世界传感器数据中的多样性和噪声，可能影响在真实环境中的表现。

NVIDIA Isaac Sim的架构

NVIDIA Isaac Sim 的架构基于 NVIDIA Omniverse 平台，旨在为开发者和研究人员提供高保真度的虚拟环境，以支持机器人设计、开发、测试和优化。通过结合 NVIDIA 的先进技术，Isaac Sim 能够创建极其逼真的仿真环境，确保虚拟测试与真实世界高度一致。以下是其架构的各个组成部分和工作流程的详细介绍：

1. 灵活的输入格式

Isaac Sim支持 Onshape、URDF、MJCF 和 USD 等多种文件格式，便于导入不同来源的设计。每种输入格式有其特定的用途和功能：

• Onshape：Onshape是一款基于云的计算机辅助设计（CAD）软件，类似于Fusion360，适合进行三维建模和设计。Isaac Sim支持直接导入Onshape模型，使得设计者可以无缝地将其CAD作品转化为可用于仿真的机器人模型。

• URDF（统一机器人描述格式）：URDF是一种基于XML的文件格式，广泛应用于ROS（Robot Operating System）中，用于描述机器人模型的几何形状、连接性、关节、传感器及其他相关信息。Isaac Sim可以通过相应的导入插件导入URDF文件，帮助用户在仿真中创建复杂的机器人系统。

• MJCF（MuJoCo描述文件）：MJCF同样是基于XML的文件格式，专用于描述多关节机器人及其他物体，主要用于MuJoCo模拟器中的动力学模拟。通过MJCF，Isaac Sim能够仿真各种复杂的动态系统，为研究提供重要支持。

• USD（通用场景描述）：USD是一种由Pixar开发的开源3D场景描述和文件格式，用于在不同工具之间进行内容创建和交换。Isaac Sim使用USD作为核心数据交换格式，这一格式的灵活性和强大功能使得在不同应用和领域之间共享数据变得更加高效。

2. 强大的处理核心

Omniverse Kit：Isaac Sim的核心构建基于NVIDIA Omniverse Kit SDK，Omniverse Kit是一个用于构建本机Omniverse应用程序和微服务的工具包，具备丰富的功能扩展能力，并支持Python脚本定制。它通过轻量级插件提供多种功能，这些插件使用C接口编写，确保API的兼容性。Omniverse Kit还集成了Python解释器，允许用户进行脚本编写和功能定制，开发新的扩展和Omniverse体验。

3. 多模式运行

Isaac Sim支持两种运行模式：

• 无头模式（Headless Mode）：此模式下，Isaac Sim在没有图形用户界面的情况下运行，通常用于自动化任务和远程计算。此模式减少了渲染负担，适合于大量并行处理和后端计算任务。

• 桌面模式（Desktop Mode）：在这种模式下，Isaac Sim通过图形化用户界面运行，用户可以直观地构建和管理仿真环境，进行交互式操作。

4. 插件化扩展

Isaac Sim的功能通过多种插件（即Omniverse Extension）实现。部分插件专用于导入不同格式的描述文件，例如URDF和MJCF的导入插件。ROS Bridge插件则用于与ROS和ROS2生态系统的通信，使得Isaac Sim能够与真实机器人进行互动。此外，Core插件则提供了Isaac Sim的基本功能，构成了其强大的核心。

5. 输出方式

Isaac Sim的输出可以通过多种途径进行：

• ROS交互：通过ROS Bridge插件，Isaac Sim能够与ROS和ROS2系统进行实时交互。这种互动性使得用户能够在仿真环境中测试和优化算法。

• Omniverse Kit Remote/Web Browser：Omniverse Isaac Sim还支持通过Omniverse Kit Remote和Web Browser进行交互，这通常是在无头模式下进行的，有助于远程访问和管理仿真环境，并在Web浏览器中查看和操作仿真场景。

• 数据集生成：Isaac Sim具有生成虚拟仿真数据集的能力，这些数据集可用于训练机器学习模型、测试算法或进行其他仿真实验。生成的仿真数据集为开发者和研究人员提供了丰富的资源，以验证和优化机器人算法。

NVIDIA Isaac Sim的工作流程

NVIDIA Isaac Sim提供了三种主要工作流程：图形化界面（GUI）、扩展（Extension）和独立Python。这三种工作流程各具特点，适用于不同的开发需求，帮助开发者在机器人仿真和开发中取得最佳效果。

1. 直观的图形化界面（GUI）

NVIDIA Isaac Sim 的 GUI 不仅提供了丰富的工具集和灵活的工作流程，更重要的是它依托于 NVIDIA 在图形处理和高性能计算领域的深厚积累，确保了所创建的虚拟环境具备极高的保真度。这种高保真度的仿真平台对于加速机器人技术的研发周期、降低实验成本以及提高最终产品的可靠性都有着不可估量的价值。Omniverse Isaac Sim 的图形用户界面 是一个功能强大的工具，旨在为用户提供直观、可视化的环境来创建、模拟和操作虚拟世界，确保了仿真的高度逼真度。其设计与 Omniverse USD Composer 相似，但针对机器人开发进行了优化。以下是该 GUI 如何实现直观性和高保真度的几个关键点：

直观的世界构建

• 高精度建模：利用 NVIDIA Omniverse 平台的强大渲染引擎，Isaac Sim 能够创建极其精细和逼真的地形、建筑物以及各种物体。这包括支持复杂的几何形状、材质属性和物理行为。

• 实时交互：通过实时编辑和预览，用户可以在构建虚拟环境时立即看到结果，从而快速迭代设计并达到所需的真实感。

• 环境细节：用户可以添加天气效果、光照变化和日夜循环等动态元素，这些都基于真实的物理规律模拟，进一步增强了仿真的真实度。

机器人组装与配置

• 精确组件：开发者可以从广泛的预定义组件库中选择，或自定义机器人的每个部分，以确保它们在视觉上和功能性上都尽可能接近实际硬件。

• 传感器集成：除了基本的机械结构外，还可以为机器人添加多种类型的传感器，如激光雷达（LIDAR）、摄像头、IMU等，并且这些传感器的行为是基于真实设备的参数进行模拟的。

• 关节动力学：仿真中的机器人能够根据现实世界的物理法则移动，包括重力影响下的运动轨迹、摩擦力作用下的速度调整等。

传感器连接与配置

• 多类型传感器：支持多种传感器的接入，每种传感器都能准确地反映其在真实世界中的工作方式。例如，相机传感器能捕捉到与真实相机相同的视角和图像质量。

• 感知能力：通过先进的算法和 GPU 加速处理，传感器数据流被实时生成，提供给机器人用于决策制定。这种数据流的质量和多样性对于训练机器学习模型特别有价值。

• 反馈机制：传感器不仅能收集信息，还能与其他系统互动，比如使用触觉反馈来模拟接触事件，增加交互体验的真实性。

ROS桥接的初始化

• 无缝集成：借助 ROS Bridge 插件，Isaac Sim 可以轻松地与 ROS 生态系统对接，允许用户直接在虚拟环境中开发和测试基于 ROS 的应用程序。

• 数据一致性：从虚拟环境到 ROS 主题的消息传递保持一致性和准确性，使得在仿真中获得的结果可以直接应用于实际部署。

• 高效调试：能够在安全可控的虚拟环境中对 ROS 应用程序进行全面测试，识别潜在问题，优化性能，最终提高产品上市前的信心水平。

2. 高效的扩展机制

扩展是建立在Omniverse Kit基础上的核心构建块，提供多种功能扩展。其设计旨在支持异步运行、实时响应及热重载功能，以满足不同的开发需求，提高开发效率。

主要特点：

异步运行：扩展能够与USD舞台互动而不阻塞渲染和物理步进，确保用户体验的流畅性。

热重载功能：此功能允许开发者在运行过程中修改应用程序代码，并即时查看更改，无需重启Omniverse Isaac Sim。这大大提高了开发和测试的效率。

事件触发机制：扩展中的操作通过事件触发的回调机制完成，这使得应用能够灵活响应不同场景和用户需求。

3. 独立Python控制

在无头模式下运行时，独立Python工作流程允许开发者对仿真过程进行精细调控，特别适合大规模实验需求。通过这种模式，开发者能够精确控制物理模拟和渲染步骤的时间，从而在无需图形用户界面的情况下高效执行任务，显著减少资源消耗。这种方式不仅提高了大规模仿真的效率，还为需要高精度控制的研究和应用提供了强大的支持。

主要特点：

高效的资源管理：独立Python模式能够在不需要图形界面的情况下运行，减少资源占用，特别适合进行大规模仿真任务。

灵活的控制：开发者可以在Python脚本中实现对虚拟世界中元素的精确控制，满足各种研究或应用需求。