

# 几何计算前沿 第一次作业

DDL: 2025.3.25

本次作业需要实现TSDF Fusion，满分15分。TSDF Fusion是Kinect Fusion的一个简化过程，其主要目的是根据相机参数、若干张深度图和对应的相机位姿，构建一个TSDF场，然后在TSDF场上使用Marching Cubes算法得到最终mesh。

## 输入数据

共1000张连续扫描的深度图，每张深度图有一个对应的相机位姿。

data/camera-intrinsics.txt:	相机内参矩阵
data/frame-000***.depth.png:	深度图，单位为米
data/frame-000***.color.jpg:	颜色图
data/frame-000***.pose.txt:	相机位姿矩阵

## Task1: 从深度图生成点云 (3 pts.)

已知相机内外参和深度图，可以根据以下公式将深度图的像素坐标 $(u, v)$ 变换到相机坐标系下的坐标 $(x, y, z)$ 得到点云。

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = z \begin{bmatrix} 1/f_x & 0 & -c_x/f_x \\ 0 & 1/f_y & -c_y/f_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

本步骤需要完成：

- 在 `fusion.py` 中实现 `cam_to_world` 函数，将深度图转为相机坐标系，再利用相机位姿转换为世界坐标系下的点云
- 对点云进行可视化（可以用 `trimesh.PointCloud` 输出为 `.ply` 文件并使用MeshLab打开）
- 在 `demo.py` 中，根据点云坐标的最大最小值，在 `demo.py` 中确定TSDF的体素场范围 `vol_bnds`

## Task2: 从深度图采样 (5 pts.)

与上一步相反，这一步我们需要把TSDF体素场的采样点向深度图上投影，然后从深度图上采样深度，为计算TSDF做准备。

本步骤需要在 `fusion.py` 中完成：

- 每个体素小立方体尺寸为 `vol_size`，按照 `vol_bnds / vol_size` 确定体素场的整数坐标，在该体素场内划分格点，作为TSDF的采样点。
- TSDF采样点的整数坐标依次转换到世界坐标系、相机坐标系、像素坐标
- 为每个落在图片范围内的TSDF采样点从深度图上采样深度

### Task3: 计算单帧TSDF (2 pts.)

根据每个采样点的深度，可以近似计算得到该点的TSDF值。

$$TSDF = \min\{1.0, (depth - z)/t\}$$

本步骤需要在 `fusion.py` 中完成：

- 根据深度计算单帧的TSDF值

### Task4: 融合多帧TSDF (3 pts.)

已知单帧的TSDF数据，我们需要将其通过加权平均的方式融合得到整体的TSDF场。

$$D_{i+1}(x) = \frac{W_i(x)D_i(x) + w_{i+1}d_{i+1}(x)}{W_i(x) + w_{i+1}(x)}$$

$$W_{i+1}(x) = W_i(x) + w_{i+1}(x)$$

本步骤需要完成：

- 在 `fusion.py` 中实现输入新图片后，对现有TSDF场进行更新
- 在 `demo.py` 中，利用所有的数据完成 TSDF 场的计算后，使用 Marching Cubes 算法（可以调包）抽取 mesh 保存并可视化（可以使用Meshlab）

### 附加任务 (2 pts.)

任选一项完成即可：

- 手动实现 Marching Cubes 算法
- 使用图片的颜色信息，实现带颜色的 TSDF Fusion

## 报告

请用文字叙述具体的算法实现过程，展示最终结果和必要的中间结果。建议在算法实现过程中多做可视化以保证各部分正确性。