

## 一、基础任务

### 方法与模型结构

参考了IGR方法(Implicit Geometric Regularization for Learning Shapes)，实现了一个基于MLP的点云重建表面模型。使用数据为点云的位置坐标和法向量，可直接从每个数据的pointcloud.npz中读取，学习一个MLP以点云在3维位置坐标为输入，输出该点的SDF值。

该方法中，损失函数的设计比较特别，由三项构成：

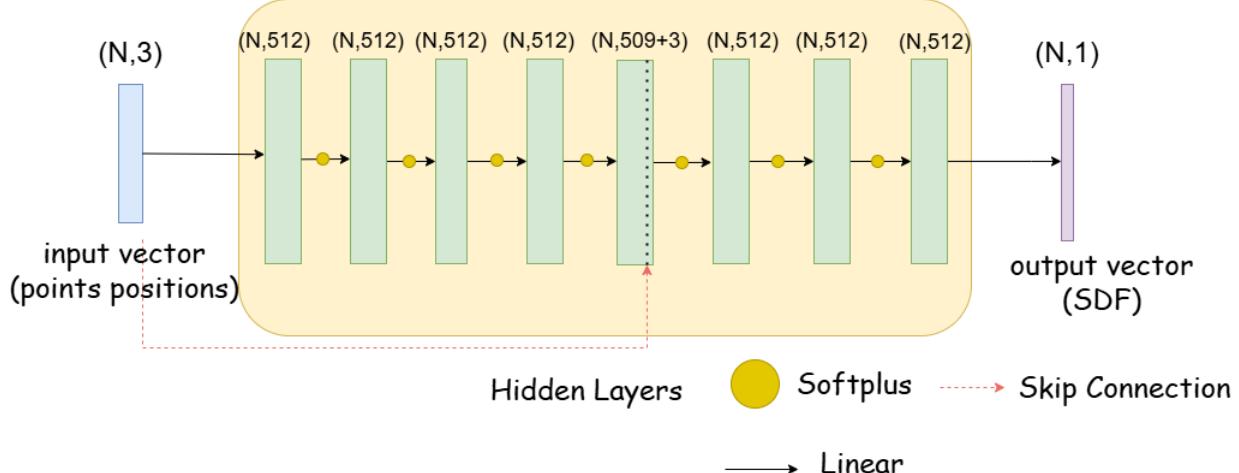
$$\mathcal{L}(\theta) = \frac{1}{|I|} \sum_{i \in I} (|f(x_i; \theta)| + \tau \|\nabla_x f(x_i; \theta) - n_i\|) + \lambda \mathbb{E}_x (\|\nabla_x f(x; \theta)\| - 1)^2$$

前两项使用每一批次中从物体表面采样的点计算，第一项约束物体表面sdf值为0，第二项约束 $f$ 在表面点的梯度与物体表面法线一致。最后一项被称为Eikonal项，强制空间中其他点梯度范数接近1，让隐式函数更接近一个距离场，使网络学习在空间中保持一致性和光滑性，这一项需要在空间中采集非表面上的点计算。

由于损失函数每次计算时需要两种不同类型点，设计了两种不同的采样方式。第一种随机采取表面点作为该批次训练数据；第二种对该批次的点进行局部扰动采集一些点，再在空间全局扰动采集另外一些点。两种点分别用于计算前两项后一项损失。对于局部扰动，在导入数据时通过构建KDTree的方式查找每一个点的第50近邻点，用他们间的距离作为局部扰动的高斯核；对于全局扰动，使用论文的参数1.8。

MLP模型结构：

## MLP Model



隐藏层有8层，维度都是 $(N, 512)$ ，相互之间通过Softplus激活函数( $\text{beta}=100$ ) 和Linear层(使用几何初始化)连接。按照论文中加入了一个skip connection，设置第5个隐藏层为skip layer。参数量约为 $3*512+512+6*(512*512+512)+512*509+509+512*1+1=1839614$ 。

训练完成后，划分一个512分辨率的网格，把所有格点坐标输入MLP，得到它们的sdf函数，再由marching\_cubes即可重建出表面。

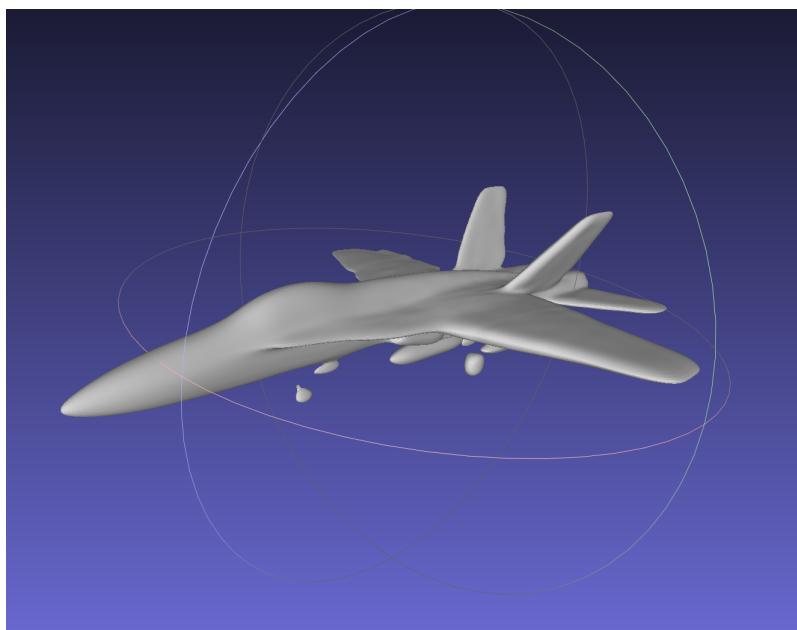
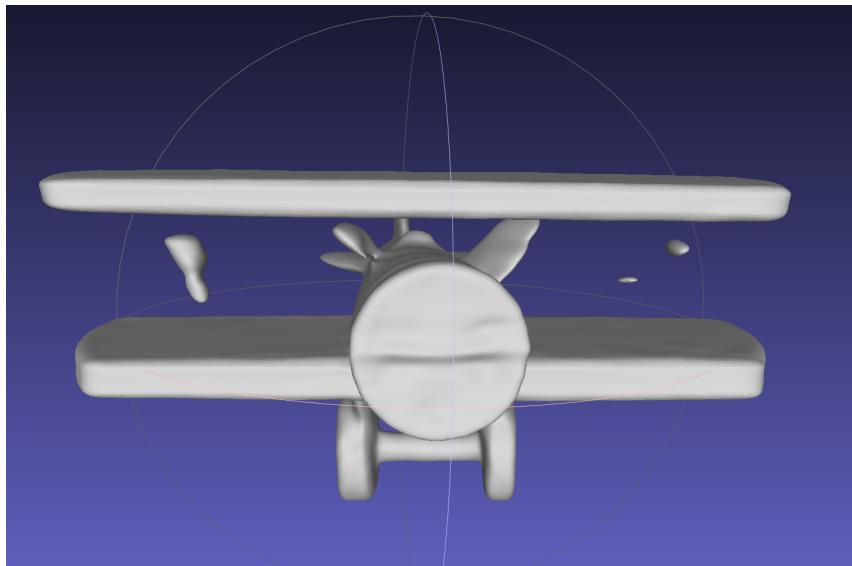
## 训练

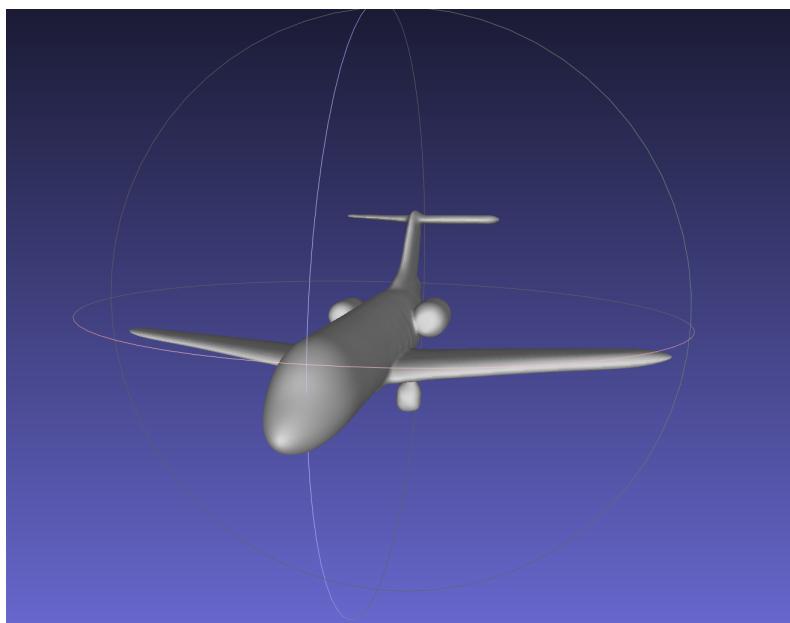
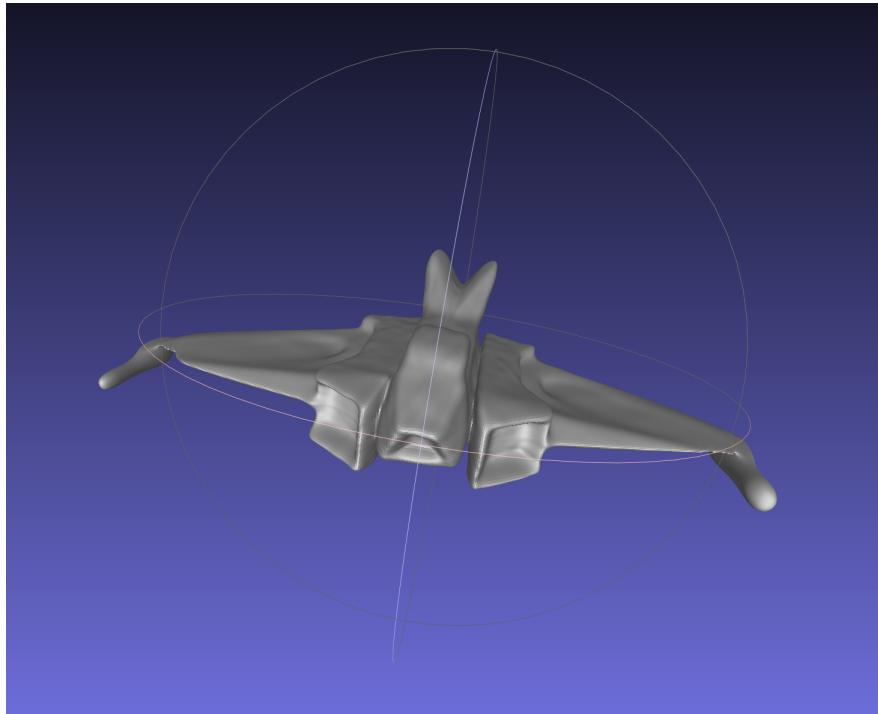
每个三维物体训练8000个epoch，其他参数沿用原论文的设置，学习率 $lr=0.001$ ，使用SGD优化器（参数为0.9），损失函数中 $\lambda = 0.1$   $\tau = 1$ ，batch size为16384，全局扰动采样时使用的 $\sigma=1.8$

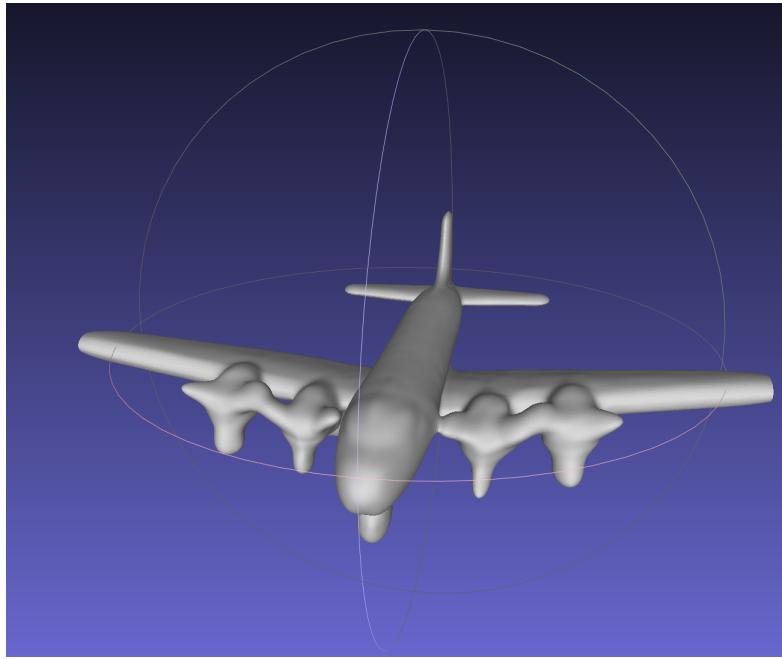
在RTX 4060 16G 上训练，训练和重建总耗时如图所示：

```
HW4 > record.txt
1  object 1 running time: 1164.2711164951324s
2  object 2 running time: 1162.5822005271912s
3  object 3 running time: 1166.2876331806183s
4  object 4 running time: 1162.8016335964203s
5  object 5 running time: 1162.999487400055s
6
```

## 结果







分析：重建的物体大体与ground truth相符，但是非常平滑，细节损失较为严重。观察到，同样的参数下，每次的训练结果也不同，大概是因为初始化具有随机性。

## 二、拓展任务

为了重建出更多的细节，将点的位置坐标经过傅里叶变换后在输入MLP，这么做是为了让模型对输入位置具有平移不变性，提高对于高频信号的拟合能力。此任务在基础任务代码上进行修改。

网络结构方面：加入傅里叶特征层，在点云位置坐标输入MLP前先进行位置编码，傅里叶频数为16，采样的标准差是5，并且傅里叶权重不需要模型训练。MLP网络深度加深为10层，跳跃连接层变成第6层，由于编码后的输入不再符合几何初始化的假设，在中间层使用kaiming初始化，在输出层采用Xavier初始化，其他设置与基础任务一致。更改后，新模型的参数量为 $3*16+16*512+512+8*(512*512+512)+512*509+509+512*1+1=2371630$

训练数据方面，除了使用pointcloud.npz读取表面点位置坐标和法向量，还使用sdf.npz读取空间采样点的坐标，sdf值，梯度。每次训练时，采样点有80%来自表面，20%来自空间中的采样点。

损失函数的设计：

$$\mathcal{L}(\theta) = \lambda_1 \mathcal{L}_{sdf} + \lambda_2 \mathcal{L}_{grad} + \lambda_3 \mathcal{L}_{eikonal}$$

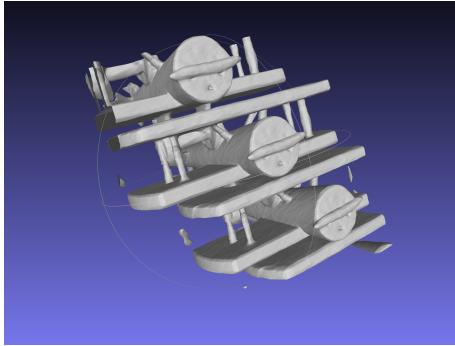
其中，

$$\begin{aligned}\mathcal{L}_{sdf} &= \frac{1}{|I_1 \cup I_2|} \sum_{i \in I_1 \cup I_2} (\|f(x; \theta)\|) \\ \mathcal{L}_{grad} &= \frac{1}{|I_1 \cup I_2|} \sum_{i \in I_1 \cup I_2} (\|\nabla_x f(x_i; \theta) - n_i\|) \\ \mathcal{L}_{grad\_norm} &= \mathbb{E}_x (\|\nabla_x f(x; \theta)\| - 1)^2, \forall x \in I_2\end{aligned}$$

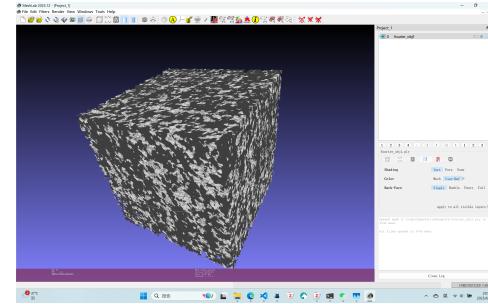
$I_1, I_2$  分别表示表面上的采样点和空间采样点。

参数方面：替换优化器为Adam优化器，学习率调低为0.0001，并且随着训练步数更新，训练80000（第一个物体）或100000（后四个物体）个epoch，每次采样30000个点，损失函数中sdf\_loss grad\_loss grad\_normal\_loss的系数分别是1, 0.1, 1。

关于参数选择的一些讨论：傅里叶频数选择过小（比如6），训练步骤太少（如10000个epoch）或者沿用基础任务的点采样方法，都会生成奇怪的结果(Figure 1)。我认为(a)的产生是因为傅里叶频数太小使得模型学不到高频知识，而是学成了简单的周期重复；(b) 的产生是因为训练不足或者空间采样点不够，导致无法消除表面之外的噪声。



(a) 傅里叶频数小



(b) 训练不足或采样点不当

图 1: 奇怪的结果

最后训练得到的结果，除了第一个物体外，或多或少的会带有一些噪声，比如(Figure 2)，为了解决这一问题，在测试时进行两个后处理：仅提取主连通块、进行适当的简单拉普拉斯平滑。

在RTX 4090 24G 上训练，训练和重建时间如图所示(Figure 3)

最终得到以下结果(Figure 4, Figure 5, Figure 6, Figure 7, Figure 8)

与基础任务中生成的物体相比结果明显更好，表面没有那么光滑，对高频细节拟合更强，物体的细节基本可以清晰展示，与ground truth更加接近。但是，美中不足的是，猜测由于损失函数中梯度项的系数比较小，梯度项损失有点大，导致重建出的表面有点粗糙，考虑

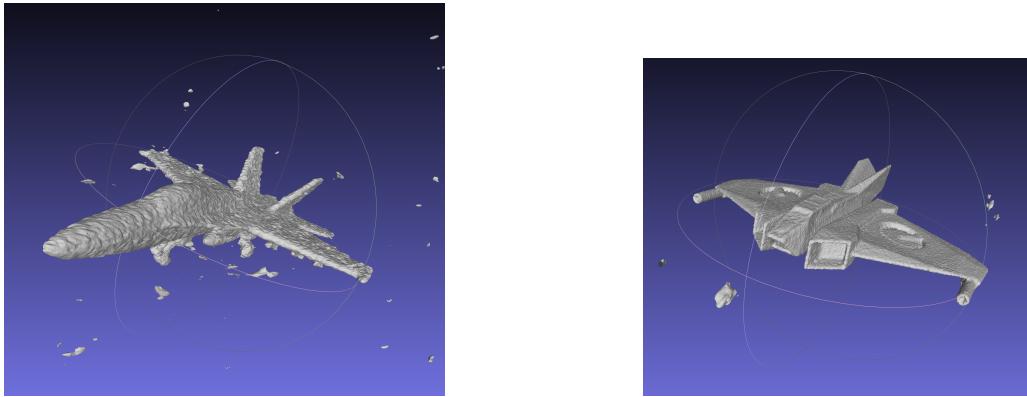


图 2: 带有些许噪声并且表面粗糙的结果

```
HW4_fourier > ≡ record.txt
1   object 1 running time: 6484.188378095627s
2   object 2 running time: 6506.524094104767s
3   object 3 running time: 6455.72102022171s
4   object 4 running time: 6522.909002542496s
5   object 5 running time: 6525.766417264938s
6
```

图 3: 训练时间

到训练成本高昂，并且现在效果已经不错，故没有重新更改参数训练。

## Reference

[IGR方法](#)

[IGR详解](#)

[几何初始化](#)

[IGR官方实现](#)

[拓展任务论文详解](#)

[位置编码与傅里叶特征](#)

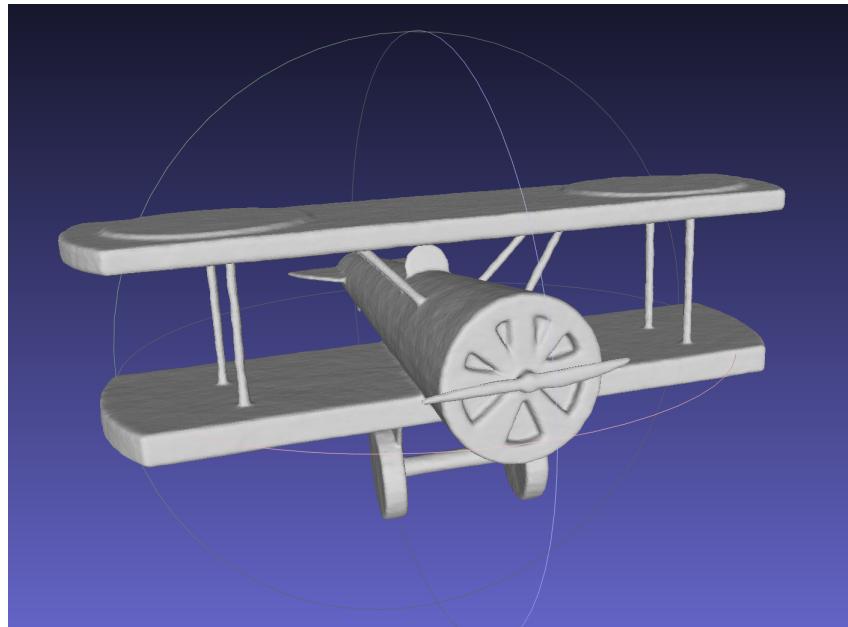


图 4: 重建结果1

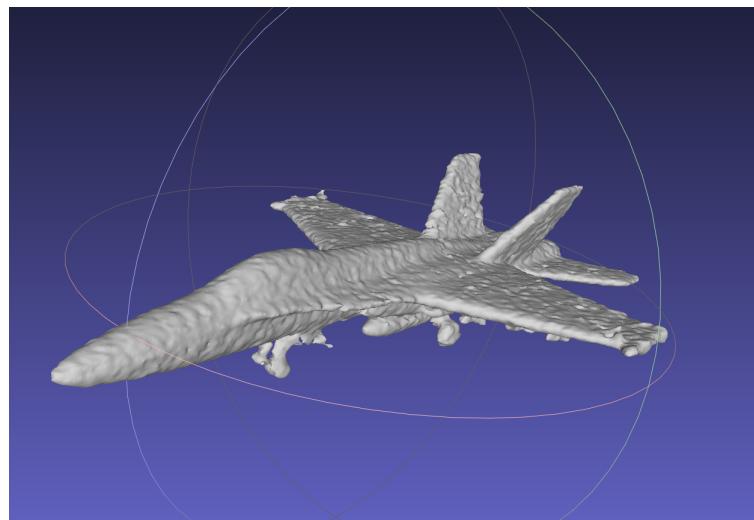


图 5: 重建结果2

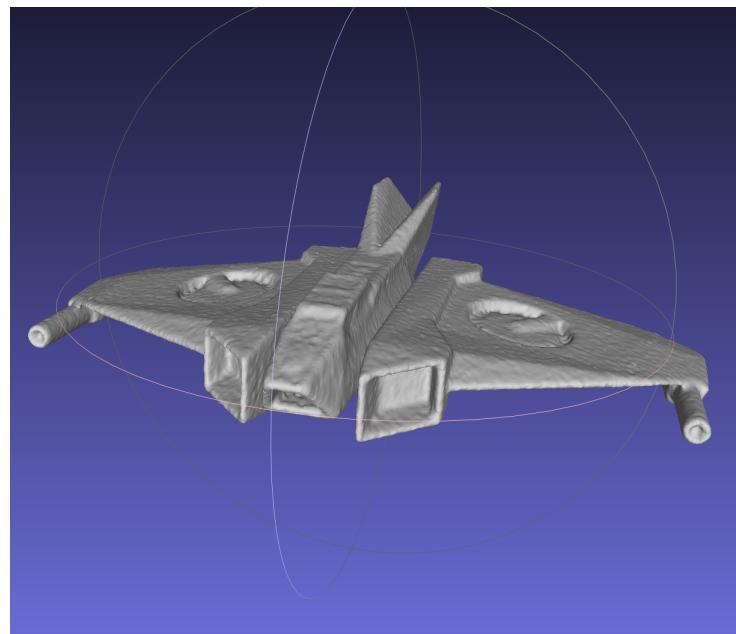


图 6: 重建结果3

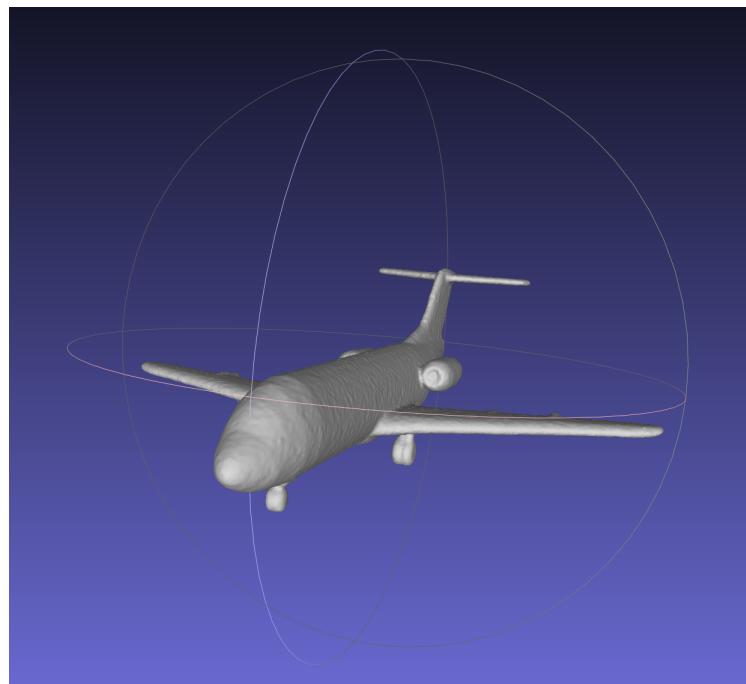


图 7: 重建结果4



图 8: 重建结果5