目录

论文思路整理 2
1. 概述 2
2. 手眼标定 2
3. 点云处理 2
3.1 技术选择 2
3.2 待解决问题 3
手眼标定方案讨论： 4
1. 标定球方案-空间定点[23-10-26] 4
1.1 尝试自动标定方法 4
2. 非空间定点 5
3. 法兰盘标定 6
4 改良 - 法兰盘标定 8
4.1 原版存在的问题 8
手眼标定实验 18
1. 基于标定板正交属性的手眼标定 – 无误差分析 18
1.1 结果： 18
1.2 关于 PSO – 慎用！ 18
1.3 结果分析 19

论文思路整理

1. 概述

首先，论文最终想要呈现的效果是一个系统，由【视觉引导的机器人系统】，名字待定。其包括手眼标定 -> 自动标定 -> 点云处理[依赖高精度手眼标定进行的配准，依赖无监督小样本的点云分割]。

为什么要进行手眼标定 –

为什么要使用深度学习的方法来进行点云处理：

传统方法不具有足够的鲁棒性，针对环境的改变没有很强的适应性。例：目前使用的点云提取方法中，噪点是是否提取成功的重要因素。

2. 手眼标定

3. 点云处理

~~首先，想要拿一个其他数据集所训练的网络直接取分割工件是不可能。得到的网络只是数据集中输入和其标签之间的映射关系，虽然大多数网络和算法都可以做到提取局部特征和全局特征，但在训练学习的过程中，是通过预测值和真实标签之间的关系建立的损失函数，来构建网络的。~~

3.1 技术选择

无监督学习 + 有监督 & 小样本 + 迁移学习 + PointNN 提速训练 + 传统方法提取工件目标区域作为标签

基于 graphTER，虽然这个网络自称是无监督网络，但是在进行解码时还需要利用标签来训练分类器(解码器)实现标签和提取到特征的对应关系。不足：在工业应用中，为工件点云打标签是不显示的。改进：利用由传统方法提取出的 ROI 作为标签

- 使用多模态，加入由点云不同的随机视角生成的 2D 图像的特征提取，然后在跨模态间进行对比学习。
- 借助 segmenta-anything
- 点云特征提取器{Point-NN，GraphTER，PointGLR}
- 图像特征提取器，应该就可以使用 segmenta-anything

【点云系列】PointGLR: Unsupervised Structural Representation Learning of 3D Point Clouds-CSDN 博客

AAAI 2022 | 一种 3D 场景多模态对比学习新方法 SimIPU - 知乎 (zhihu.com)

CVPR 2022 | CrossPoint：3D 点云理解的自监督跨模态对比学习 - 知乎 (zhihu.com)

3.2 待解决问题

1. 如何处理噪点问题
2. 网络部署在哪里，部署在何时 -> 测试阶段，在系统开发阶段就部署网络，利用有监督学习生成的网络和测试阶段的小样本数据进行微调，样本标签可使用传统方法提取出的目标点云。
3. 针对有监督学习网络的训练，应该多多贴近实际项目需求，目前大多数点云分割任务所在做的是将一个点云的不同部件全部分割出来。而在实际项目中，我们只需要将目标区域的点云分割出来即可！

   以 shapeNet 为例，其将飞机主体、机翼、尾翼等全部分割，其与我们要处理的任务并不完全相同，可能我们只需要飞机的尾翼，而其他部位是否分割成功，我们并不关心。那么在进行网络设计时，我们可能需要圈定 ROI，然后寻找点云与 ROI 之间的关系。

   但是这样看起来好像并没有把任务简单化。原任务是把点云进行拆分，而现在的任务变成提取指定 ROI。

手眼标定方案讨论：

1. 标定球方案-空间定点[23-10-26]

硬件：标准球

标定原理：扫描标准球，通过圆拟合求出截面的圆心，根据勾股定理求出标准球球心，进行${}^{B}_{C}T{}^{C}_{O}P={}^{B}_{E}T{}^{E}_{O}P$ 的坐标转换，根据误差方程$\Delta_i=Ab_i-c_i$ 求出手眼矩阵 A。

改进方法：根据线激光相机光学成像原理，激光线在物体表面的高度差越大误差越大，所以在标定过程中改变位姿只关注拟合圆半径最大的地方，越接近球体半径效果越好。存疑：后续应进行相应的实验进行误差分析。

最优化方程-1：$\Delta=\sum_i \Delta i$，（$\Delta_i=A{}^{C}_{O}P_i-{}^{B}_{E}T_i{}^{E}_{O}P_i$）

主要误差：

- 硬件误差：${}^{E}_{O}P$，标定物的硬件误差。
- Δ的最优化误差

最优化方程-2：利用标定物和法兰盘的相对不变。（${}^{E}_{B}T{}^{B}_{C}T{}^{C}_{O}P$）$_1$=（${}^{E}_{B}T{}^{B}_{C}T{}^{C}_{O}P$）$_2$=……

主要误差：

- 相较于上式，减少了硬件误差，两者最优化方程的误差比较需进行实验尝试。

1.1 尝试自动标定方法

- 已知${}^{E}_{O}P$ – 最优化方程 1：

  随便将标定球移至激光线下(尽量靠近球心位置)，然后开始自动标定。

  标定程序：第一步，求解粗糙精度手眼矩阵，首先拟合当前激光线下的圆，求出圆心坐标，然后求出球心坐标。调整机器人位姿，使机器人重新到达距离球心坐标 5%的误差范围内，根据求得位姿结果求解手眼矩阵。利用手眼矩阵求得当前拟合圆心在基坐标系下的坐标位置，然后将球心移至此坐标位置。然后控制机器人以不同位姿到达此空间定点，然后求解手眼矩阵。

- 已知${}^{E}_{O}P$ – 自行建立外部基坐标系 – 最优化方程 1：

  利用 tcp 在激光线处建立基坐标系(x，y 轴方向尽量与相机坐标系保持一致)，然后安置标定球，将标定球球心移至外部基坐标系原点。拟合圆，根据半径大小，判断需要沿 y 轴哪个方向运动，然后找到最大半径，确定此定点位置坐标，然后改变机器人各轴位姿重新达到此位置坐标(可再次进行最大半径确认)。

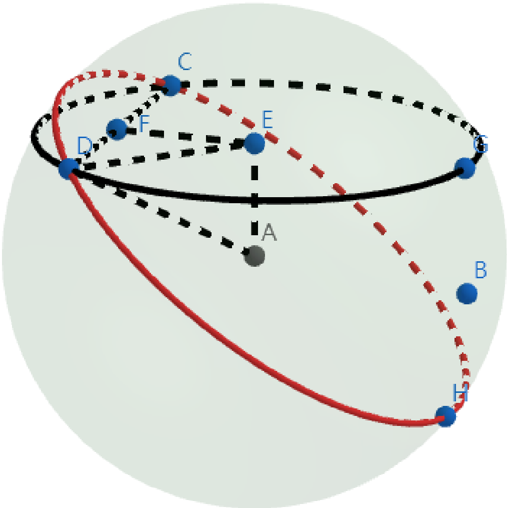
- 未知$^{E}_{O}P$– 自行建立外部基坐标系– 最优化方程 2：效果也许比使用最优化方程 1 好

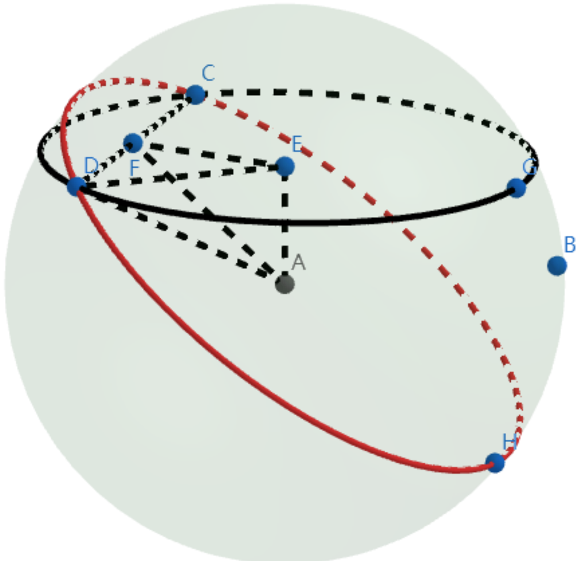
2. 非空间定点

非空间定点的方法，例如标定点，需要求出相机的外参及标定物相对于相机的位姿，采用空间定点的方法理论上比非空间定点的方法要多一些求解步骤。

3. 法兰盘标定

核心思想：利用现有的高精度机器人末端法兰进行标定，减小相机扫描数据因高度差造成的误差，消除点云拟合求解空间定点误差，消除手眼标定标定物成本。





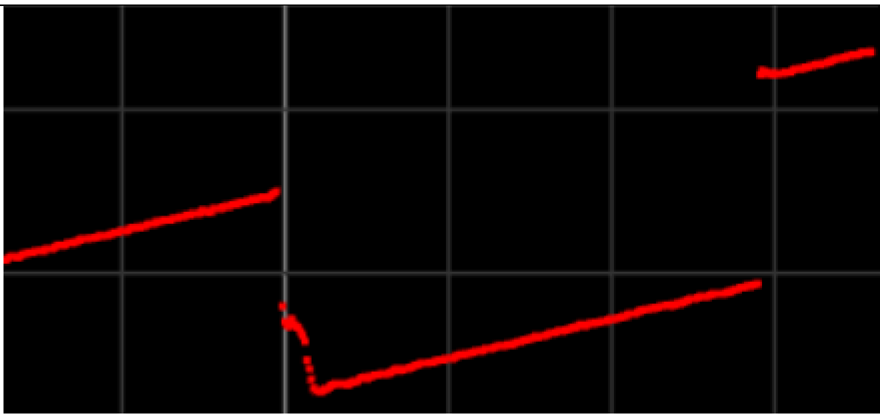
激光线与法兰相交的弦的中点 F(x, 0, z)

激光线所在平面在球中的截面圆圆心 E(x, 0, z')

以法兰为过球心的截面圆的球的球心 A(x, y, z')

以上，在三角形 DEF 中有$(z'-z)^2+DF^2=DE^2$，三角形 AEF 中有$(z'-z)^2+y^2=AF^2$，三角形 ADE 中有 $DE^2+y^2=R^2$， 以 AEF 以及 DEF 为底的三棱锥体积相同

实验中发现相机扫描轮廓线有精度问题存在，对于 CD 两点的获取存在误差。



拟解决方案：获取跳崖部分中处于低位的边界点，并考虑加上一个与高位点和地位点相关的微小值。

求解部分无法实现联立等式求解，也只能使用最优化实现求解。

虚拟球心 y 轴和 z 轴的正负问题，需要手工标记

类比误差来看，方法至少与标定球相当，但减少了成本

误差分析 1. 弦上三点坐标确立 2. 联立三个勾股定理构建方程式求解误差 3. 手眼标定求解误差	相较于标定球 1. 因高度差导致的相机扫描误差 2. 拟合圆带来的误差 3. 手眼标定求解误差

4 改良 - 法兰盘标定

4.1 原版存在的问题

约束条件不够，建立的方程无法确定唯一球体和截面圆(此时球心[圆心]，球半径[圆半径]都不是唯一的)

改善：将弦两端点的点约束在相机坐标系一球体上[不约束在圆上是因为在三维空间中不易确立圆的方程进行约束](此时球心[圆心]不是唯一，球半径[圆半径]唯一的)

此时可以求出这条弦所在球体的球心，但不唯一

可以发现这个球心都约束在以弦中心为圆心，球心到弦中心为半径的圆上。同样的，无法做到将其直接通过圆的方程进行约束，此时可以通过球的方程，将这些球心(法兰坐标系下的空间定点)约束在以弦中心为球心，球心到弦中心为半径的球上，同时对约束在球上的点做拟合圆[进行纠偏]。

原 PSO 求解的是 y, z'和 DE，从上述的分析可以得出这样的模型存在很大的问题

1. 默认 E 点及截面圆的圆心与所求球心的 Z 相同，显然不同

2. 默认 E 点和 F 点 y 相同，显然不同

即此模型中只有三个三角形的边长相同，其他且为变量

所以将 PSO 待求解值改为 [A['y'], A['z'], E['y'], E['z']]

使用两点两点间的举例表示边长进行求解

求解精度以 DE 长度为判断标准

不行，点 A 和点 E 的 x 也是位置的，但是把两个 x 设置为变量的话无法使用最优解求解。

考虑到因为边 AE 的计算全是变量，精度会很差，所以把 AE 也设置为变量

Best_y 是 inf 的主要原因是原来设置的约束条件有问题

- 约束条件要带有 partical，要不然你约束了个寂寞啊

想办法对点 E 进行约束，还是利用球的方程

除了最优化方程，其余的约束条件应该使用约束函数进行约束，不能直接加在最优化方程上：

1. A = {'x': particle[0], 'y': particle[1], 'z': particle[2]}

   E = {'x': particle[3], 'y': particle[4], 'z': particle[5]} 作为需要求得解

2. 约束 AD- R <= 0　AD- R <= 0(如果将 C、D 约束在球的方程上，那么这个约束条件

优点多余)

3. 将点 C 点 D 约束在以点 A 为球心的方程上

4. 对点 E 约束 具体的 DE < AD | AE < R

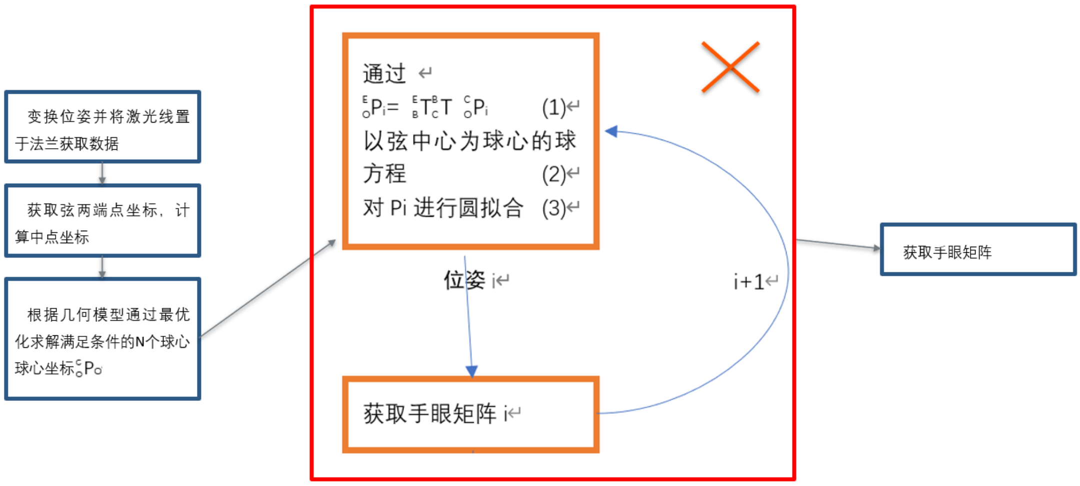
5. 应该再加一个将点 C，点 D 约束在以点 E 为圆心的圆上，具体的 CE-DE <= 0

因为没有办法在三维空间中建立圆的方程，所以只能这样对点 E 进行约束

利用法兰盘内圆进行标定时，部分位姿会遮挡，无法求解出圆上的点：

这样不利于确立激光线与圆相交弦的两个端点 -> 使用大外圆(40mm 的)可以直接判断使用落差处的点[亦或者使用其对应的小外圆]

方案 3 中进行的误差分析	误差分析
1. 弦上三点坐标确立 2. 联立三个勾股定理构建方程式求解误差 3. 手眼标定求解误差 4. [遗漏]当法兰盘斜对相机时仍然会存在由高度差带来的相机误差	➔ 激光线成像不理想 ➔ 取点有误差，方程不一定有解，所以要使用最优化求解最优解 ➔



5 球心求解优化思路及后续操作

5.1 球心求解优化

为实现 - 提高精度优化：求解出的球心应该在同一平面上的同一个圆中，以球心到平面的

距离以及球心所拟合出的圆为优化方程进行求解。
为实现 – 提高算法速度：修改重复无必要约束条件，修改粒子群参数找到最均衡的参数列表

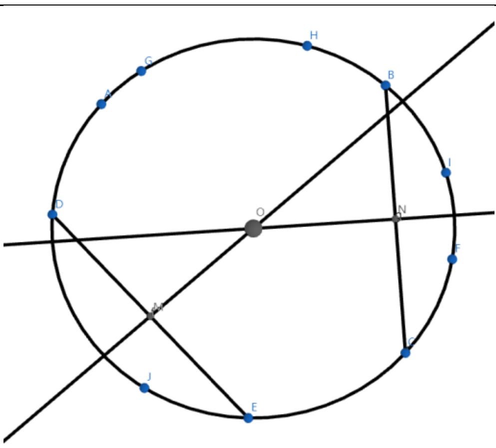
5.2 手眼矩阵求解

5.2.1 方法 3、4 的误区

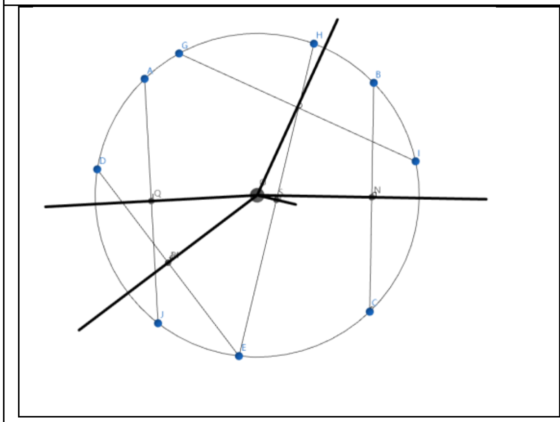
每一组位姿确定一组空间定点，可以求解出一个手眼矩阵。	求解出的一组点，准确来说并不是空间定点，它们只是具有相对位置关系的一组点。因此，对于这组点的相对位置关系，存在一个问题，即存在任意的齐次变换矩阵，可以将它们在三维空间中进行变换，而不改变它们之间的相对位置关系。

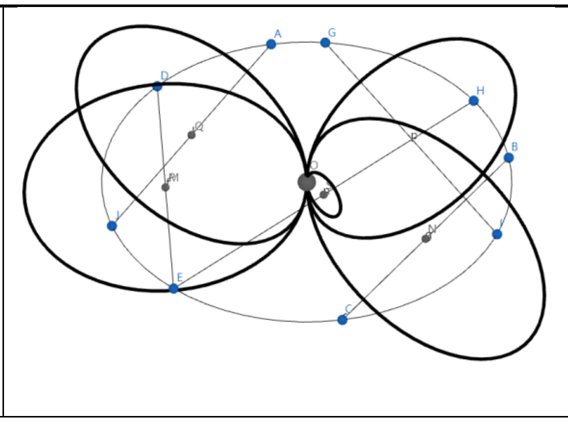
5.2.2 分析手眼矩阵计算思考过程

这个方法中存在的唯一定量： 法兰盘圆心　（这个可以根据两条弦的垂直平分线的交点获得[但是只能在法兰盘坐标系下求解]） 根据上述定量可获得： 过法兰盘圆心的法兰盘法向量
约束条件[最优化方程]： 1. 根据两条弦的垂直平分线的交点求得点 O

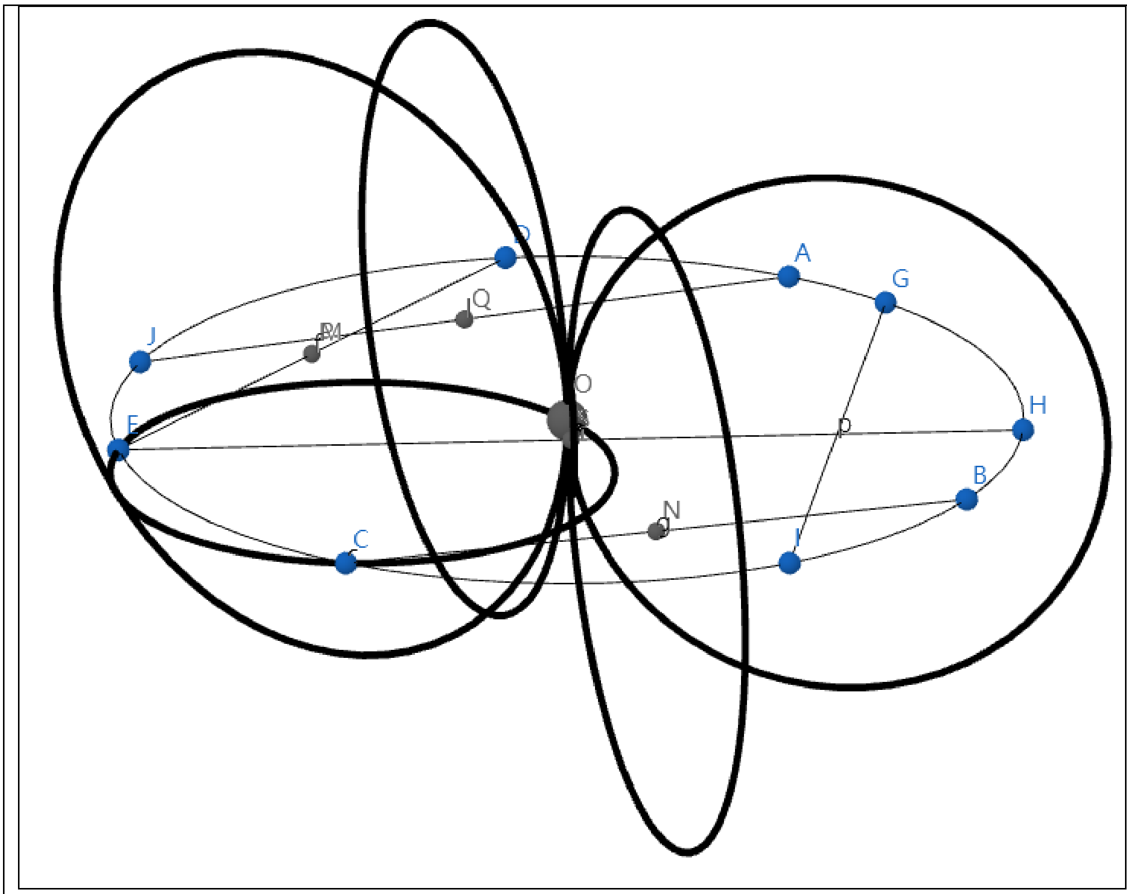


2. 各球心圆相较于一点 A[法兰盘圆心]

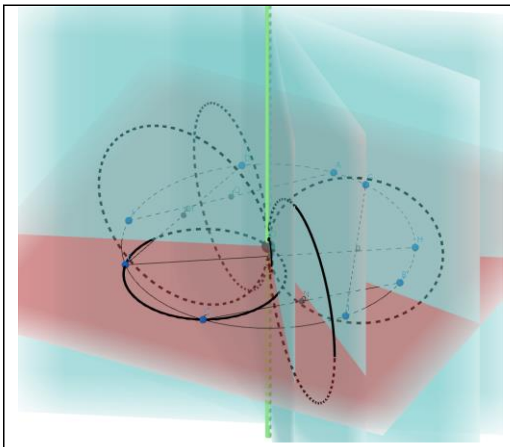


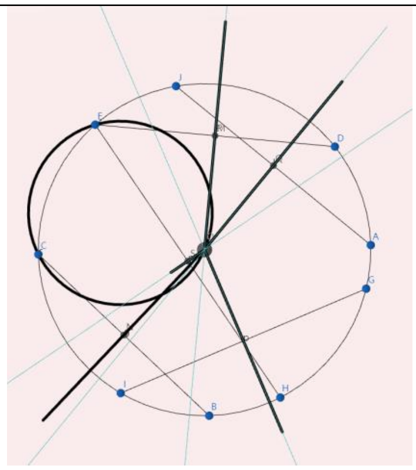


3. 在进行手眼标定求解时，我们将一组球心进行空间变换[也就是这组球心组成的圆，那么存在以下情况，满足 1，2 约束而不是理想结果]



4. 此时就要对这些球心所在的平面进行约束[所有平面相较于同一条直线，且为过点 O 的法兰盘平面法向量所在的直线]





此时可实现求解唯一的理想的手眼矩阵。

上述 – 总结:

最优化参数手眼矩阵和点 O 坐标

1. 各圆相交于一点 O[各圆心到点 O 的距离等于圆半径]

2. 各平面相交于过点 O 垂直于法兰平面的直线[各平面要过此直线，且各平面交线到其距离为 0]

3. 验证误差，通过中垂线相交求解点 O'与最优化求解结果相比较

4. 最优化方程：各弦端点约束在以点 O 为圆心，半径为法兰半径的圆上

********** 三维空间中不好确定圆的方程，可以将点先约束在圆所在的平面上，然后计算点到圆心的距离

完整步骤：

1. 通过圆上的弦(激光弦)与一组几何关系和约束条件得到一组约束在以弦中点为圆心，弦到法兰圆心的距离为半径的圆上的球心，其中包含我们模拟的球的球心 O[即法兰盘圆心]

那么传入下一步进行手眼矩阵求解的输入为：

Array[N][] = {{平面方程， 圆的方程， 圆的半径， 圆心， 弦两个端点} ……}

通过两个端点即可算出平面方程：$a(x-x1) + b(y-y1) + c(z-z1) = 0$, <abc>为方向向量，$(x1, y1, z1)$为上面的点

2. 将手眼矩阵和法兰圆心 O 最为最优化求解参数

上单元格 2 中的约束不易实现，可以做如下更改：

约束各平面法向量(弦)与过点 O 的法兰平面法向量垂直，然后约束点 O 到平面的距离为 0

上述步骤存在漏洞，以下是新的想法：

1. 在法兰盘坐标系下，一个位姿对应的 N 个球心组成的圆所在的平面 A 是垂直于法兰盘平面 B 的，且弦是这个平面 A 的法向量，那么也就是说

2. 在法兰盘坐标系下，所有弦的端点都应该共面，即法兰盘平面。可以使得每两个端点+法兰中心确定的一个平面，组成的平面集中所有平面两两平行(约束 1)，且法兰中心到端点的距离等于法兰半径(约束 2)，实现将变换后的弦端点约束在法兰坐标系下的法兰平面上，且是个圆。

那么需要求解的最优化变量就是手眼矩阵和法兰中心

在约束 1 的情况下，两两平面之间的距离应该都为 0，此为约束 3

所有平面应该相较于一条直线，且这条直线是过法兰中心的法兰平面的法向量，此为约束 4

终极答案!!!!!!

这个手眼标定模型存在一个可以求解的某一定量，就是使用到的法兰平面!!! 是唯一的!!

所以上面提到的约束 3 应该作为最优化方程，去求解唯一的平面，在此基础上约束法兰中心和弦端点。

5.2.3 最终方案 0.1

<table>
<tr><td colspan="2">变量定义：
● 法兰中心 O
● 第 i 个位姿下球心所在平面 M_i、法兰平面 N_i
● 弦的两个端点和中点 C_i、D_i、F_i
● N_i 平面利用向量 C_iD_i 和点 F_i 表示</td></tr>
<tr><td>几何关系：</td><td>代数运算：</td></tr>
<tr><td>法兰中心 O 位于所有法兰平面 N_i 上</td><td>~~约束法兰中 O 到所有法兰平面 Ni 距离为 0~~
~~约束空间一定点到所有法兰平面 Ni 距离为 0~~
约束 Ci Di 到其他平面 N 的距离相等</td></tr>
<tr><td>所有法兰平面 N_i 应该同时表示法兰坐标系下的固定平面-法兰盘平面</td><td>需要分两步将集合 N 中的平面约束到法兰平面
1 通过三点 O、Ci、Di 确定 Ni 平面方程，进而求解平面法向量，约束所有的 N_i 平面平行，即约束其法向量平行，也就是法向量两两之间夹角的余弦值为±1
~~2 然后约束点 O 到 N_{i+1} 平面之间的距离为 0 【或者约束空间中一定点到所有 Ni 平面的距离相同】~~</td></tr>
<tr><td>平面集合 M 和平面集合 N 中的元素两两垂直</td><td>~~平面 N 的法向量和平面 M 的法向量垂直，也就是法向量两两之间点积为 0~~
理论上平面 M 的法向量 CD 在平面 N 内，所以上面这个约束等同于，将 CD 约束在平面 N 内</td></tr>
<tr><td>弦端点在以点 O 为圆心，落在法兰盘平面的圆 A 上</td><td>分为两步
1 弦端点到平面 Ni 的距离为 0
2 弦端点到点 O 的距离为法兰半径</td></tr>
<tr><td>平面集合 M 中两两平面全部交于，过点 O 的法兰平面的法向量</td><td>分为两步
1 平面 Mi 的法向量平行，也就是法向量两两之间夹角的余弦值为±1
2 点 O 到所有平面的距离为 0</td></tr>
<tr><td>点 O 应该位于以弦中点为圆心的圆上</td><td>点 O 到弦中点的距离等于 AF</td></tr>
</table>

计算思路
最优化方程使用几何关系 1，最优化求解手眼矩阵和法兰中心 O
几何条件 2 计算即约束 求解 2.1 利用 Ci、Di、O 求解法兰平面 Ni 方程
求解 2.2 求解法兰平面的法向量 约束 2.1 法兰平面法向量平行，即向量夹角余弦绝对值为 1 ~~约束 2.2 空间定点到法兰平面距离相等~~ -> 最优化方程 几何条件 3 计算即约束 计算 3.1 求解平面方程 Mi 计算 3.2 计算 Mi 法向量 约束 3.1 Mi 法向量与 Ni 法向量点积为 0 几何条件 4 计算即约束 约束 4.1 ~~弦端点到法兰平面 Ni 距离为 0~~ -> 最优化方程 约束 4.2 弦端点到 O 的距离为法兰半径 几何条件 5 计算即约束 约束 5.1 平面 Mi 法向量平行，即向量余弦绝对值为 1 约束 5.2 点 O 到平面 Mi 距离为 0 几何条件 6 计算即约束 约束 6.1 OF=AF

5.2.4 最终方案 0.2

0.1 将点 O 塞入待求解的最优化解集中，经实验发现在处理 Ni 时存在问题 现尝试不将点 O 作为待求解值
1. 没有了点 O 如何确定 Ni 使用两两 CD 点组构建 Ni 平面集
2. 约束条件

6 手眼标定 Bug

6.1 **欧拉角求解

```
# 获取欧拉角并转换为旋转矩阵 默认是弧度制!!!
        R_itemPose = R.as_matrix(R.from_euler("xyz", itemPose[3:], degrees=True))
```

指定 degrees=True 使用角度制

```
R_itemPose = R.as_matrix(R.from_euler("ZYX", itemPose[3:], degrees=True))
```

1. 机器人是按照 zyx 旋转的，
2. 大写表示外旋(绕固定坐标系轴)，小写表示内旋(绕动态坐标系[上一次旋转过后的坐标系])

手眼标定实验

1. 基于标定板正交属性的手眼标定 – 无误差分析

1.1 结果:

对于正交结果没有取绝对值，并且-1 操作在了总和上，而不是正交结果上

[0.35916387647123915, -0.16341298928510636, 0.14733343822295228, 0.09098256646261173, 2052.058693479467, -581.7415735667876, 578.7794487773392] 31.60478334478031

[0.522084587820512, -0.8234775924389033, 1.0157965448543802, -0.5170586066122033, 1302.4908465351934, -2161.112279317044, 1.5715959684077796] 243.1123204376505

[-0.2854214179659117, 0.13447477645743564, 1.2918097686678305, 0.44229037379147157, 2749.4538316265603, 778.6709631111136, 2080.9564076019246] -0.01057838886273288

修正后的结果(只使用了前两组数据)

```
    [-0.1956222937825032, -0.766691234111081, -1.3069606486078362,
0.4611925079381167, -18.499284071773456, 137.47627475210075,
1130.7134499266062] 7.80849483352845
```

第一组数据

```
    [-0.0991413135466297, 0.2990517623102416, 0.4023198355907962,
0.20760125872645924, -34.83984877057516, 166.67872691599973,
1160.3935743510986] 3.8762244552348335e-05
```

修正：对机器人位姿求逆

```
    [-0.3303837255305453, 1.776886984456582, 1.7709967204580646, 0.8,
-53.70164663739292, 142.53531107860917, 1145.8717964358702]
0.00010950245283147808
```

1.2 关于 PSO – 慎用！

基于空间定点的手眼标定方案还没有想好初值的问题， 但是 PSO 的求解范围对求

解的精度提升有非常大的帮助。

1.3 结果分析

最后呈现的结果很差，通过求解出的手眼标定矩阵，将标定板上的点转换至机器人末端后，进行直线拟合，发现任意两条直线都没在一个平面。

原因尚未找到。