目录

[关于写论文相关 3](#_Toc149742044)

[开题和书写基本逻辑 3](#_Toc149742045)

[博客\_网络资源 4](#_Toc149742046)

[手眼标定 4](#_Toc149742047)

[相机标定 4](#_Toc149742048)

[点云处理 4](#_Toc149742049)

[深度学习\_论文 4](#_Toc149742050)

[Pointnet & Pointnet++ 4](#_Toc149742051)

[深度学习\_技术\_概念 4](#_Toc149742052)

[maxPooling 4](#_Toc149742053)

[0. 通用问题 6](#_Toc149742054)

[0.1 坐标系转换计算 6](#_Toc149742055)

[0.2 齐次矩阵-位姿 6](#_Toc149742056)

[0.3 手眼标定矩阵等式求解 6](#_Toc149742057)

[0.4 手眼标定矩阵等式 6](#_Toc149742058)

[0.5 手眼标定中的思想-坐标系转换关系中的思想 8](#_Toc149742059)

[0.6 齐次变换矩阵及其使用（原理不太懂，） 9](#_Toc149742060)

[0.7 三维空间中的向量 – 旋转矩阵 10](#_Toc149742061)

[1.⭐误差分析(巧妙思想)-机器人激光检测手眼标定误差分析及优化 11](#_Toc149742062)

[1.1 硬件设备信息 11](#_Toc149742063)

[1.2 误差来源分析 12](#_Toc149742064)

[1.3 手眼关系及误差表达式 12](#_Toc149742065)

[1.4 \*球心拟合误差（需要看一下论文3中线激光结构光相机的成像原理） 13](#_Toc149742066)

[1.5 \*手眼矩阵计算误差及改进 14](#_Toc149742067)

[1.6 其他 14](#_Toc149742068)

[2.⭐0.0815(高精度)-基于标准球的机器人手眼标定方法改进研究 15](#_Toc149742069)

[2.1 设备信息 15](#_Toc149742070)

[2.2 手眼关系及标定流程 16](#_Toc149742071)

[2.3 其他 17](#_Toc149742072)

[3.学位论文-基于3D机器视觉的工业机器人跟踪涂胶系统\_陈琳 18](#_Toc149742073)

[4. 线激光器的手眼标定方法 19](#_Toc149742074)

[4.1 设备信息 19](#_Toc149742075)

[4.2 其他 19](#_Toc149742076)

[5. Pointnet 20](#_Toc149742077)

[5.1 解决的问题 20](#_Toc149742078)

# 关于读&写论文相关

## 结构

|  |
| --- |
| 1. 摘要   (所研究领域的)**大背景 & 问题大类**  -> (本文主要解决了哪方面问题) **当前的主要挑战**  -> **主流做法 & 存在的问题 & 瓶颈**  **-> 本文提取的方法**  **-> 方法创新点及结果**   * 2. (E BT(i+1))-1×E BTi×B C Ti = B C T(i+1)×C OT(i+1)×(C OTi)-1 X = B C Ti = B C T(i+1)       AX=XB  RARX = RXRB  (RA-I)TX = RXTB-TA  (E OT)i - (E OT)i+1 |
|  |

## 开题和书写基本逻辑

|  |
| --- |
| 1. 看论文，最起码要了解的多一些，最终呈现出来的要覆盖广！   其次，还有论文分布在哪些期刊，哪些时间点，这些可以去支撑可行性分析，同时可以窥见当前的研究现状、近几年来的研究进展和未来几年大概的发展方向和趋势。权威性！！   1. 论文肯定是解决问题的，从现实的小问题开始，解决之后解决大问题，问题越来越复杂，内容越来越丰富。 |

## 读论文

|  |
| --- |
| [谈一谈CS.CV方向如何阅读论文？ - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/396064894) |
| 以后记录的最基本架构可参考如下：   1. 解决了什么问题 & 背景 2. 优点(为什么可以做到) & 主要贡献 3. 不足(有待改进的地方) |
| 先BFS速读一遍论文，Abstract和Intro要细，Related work可以快速过甚至不看，Approach稍快，关注一些特殊模块，experiment图一乐。总体来讲第一遍要关注图和表，这样可以对文章形成初步的认识，并且判断是否值得你全文细读，有时可能只需要关键部分细读，就不必看全文。 |

# 博客\_网络资源

## 手眼标定

### 相机标定

|  |
| --- |
| [相机标定目的及原理\_相机标定的目的-CSDN博客](https://blog.csdn.net/weixin_51557361/article/details/122121505)  像素坐标系->物理坐标系->相机坐标系->世界坐标系  前两步的矩阵相乘的结果合称为相机内参  最后一步成为相加外参，也就是我们需要的标定物相对于相机的位姿。  这个所谓的标定物相对于相机的位姿不是必要条件，因为我们有标定物在相机坐标系中的坐标，以眼在手外为例，完全可以使用  E OP=E BT·B CT·C OP |

## 点云处理

|  |
| --- |
| [【3D点云】分割算法总结（一）\_3d点云分割-CSDN博客](https://blog.csdn.net/qq_45752541/article/details/126330335?spm=1001.2101.3001.6650.4&utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7ERate-4-126330335-blog-113572662.235%5Ev38%5Epc_relevant_anti_vip&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7ECTRLIST%7ERate-4-126330335-blog-113572662.235%5Ev38%5Epc_relevant_anti_vip&utm_relevant_index=9) |

## 深度学习\_论文

### Pointnet & Pointnet++

|  |
| --- |
| [3D分类与分割之PointNet 论文笔记 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/73086704) |

## 深度学习\_技术\_概念

### maxPooling

|  |
| --- |
| [对Max Pooling的理解\_maxpooling-CSDN博客](https://blog.csdn.net/weixin_41513917/article/details/102514739) |

# 通用问题

## 坐标系转换计算

|  |  |  |
| --- | --- | --- |
| [（一）关于手眼标定理论相关的笔记\_boss-dog的博客-CSDN博客](https://blog.csdn.net/qq_45445740/article/details/122170029)  结论：   |  |  | | --- | --- | |  |  | |

## 齐次矩阵-位姿

|  |
| --- |
| C OT是标定物坐标系到相机坐标系的齐次转换矩阵，实际含义是标定物在相机座标下的位姿，即以相机坐标系为基坐标系，标定物的位姿（标定物的坐标系）  B OT = B ET×E CT×C OT |

## 手眼标定矩阵等式求解

|  |
| --- |
| 求解手眼标定矩阵等式本质上就是最优化问题，因为列出等式的因式全部都是测量值，其在理论上是相等的，但是实际中是不可能完全相等的，所以涉及到了最优化问题，选择最优解。目前论文普遍使用最小二乘法进行简单筛选，目前无法确定使用最优化算法进行求解是否会带来精度的提升。 |

## 手眼标定矩阵等式

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| * 最常见的莫过于AX=XB，就是把 (E BT×B C T×C OT)i = (E OT)i = (E OT)i+1 = (E BT×B CT×C OT)i+1中的手眼矩阵设为X，等式两边同右乘C OT-1，同左乘E BT-1即可。 * 使用空间顶点方法（标准球）的论文中更多使用   PB = B ET × E CT × PC 然后在不同位姿下测量PBi 例：   |  | | --- | |  | |  |   然后使用四元数表示X(手眼矩阵)联立方程直接求解。   * 也可以将PB = B ET × E CT × PC转换为Ax=b的问题使用非线性的方法进行求解。（最优化问题）  |  | | --- | |  | |  |   旋转矩阵为正交矩阵，以其向量为单位向量为约束条件。   * 除这些常规方法以外，还可以视情况而定去表示手眼关系等式。  |  | | --- | |  |   此例为手在眼外，且文中称可以根据标定物测量得出E OX。 |

## 手眼标定中的思想-坐标系转换关系中的思想

|  |
| --- |
| * 首先是坐标系转换关系中的思想，为什么我们要做手眼标定，因为我们的手和眼不在一个维度（确切的说是不在同一个坐标系），那么我们就没有办法去让手干眼所看到的事情。所以我们要做手眼标定，即坐标系转换，而这个系统内部组成并不是静止不动的，也就是说针对每一个内部组成，其自身的坐标系原点有可能是在变化的，我们无法对其进行坐标变换。所以在进行坐标变换时，我们要找到两个系统内部相对静止的坐标系，一个是我们的目标坐标系转换矩阵即手眼矩阵，一个是用于罗列坐标转换矩阵等式用于求解。**【将变量化解，使用定值求解】** * Eye-to-hand:以眼在手外为例，相机和机器人基底的坐标系以及标定物和法兰盘坐标系明显相对静止，针对不同位姿有以下方程式：(E BT×B C T×C OT)i = (E OT)i = (E OT)i+1 = (E BT×B CT×C OT)i+1。   B->E可有机器人示教器读取，O->C可由相机读取，由此可求手眼矩阵即C->B的坐标系转换矩阵。   * Eye-in-hand:以眼在手上为例，相机和法兰盘的坐标系以及标定物和机器人基底坐标系明显相对静止，针对不同位姿有以下方程式：(B ET×E C T×C OT)i = (B OT)i = (B OT)i+1 = (B ET×E CT×C OT)i+1。   B->E可有机器人示教器读取，O->C可由相机读取，由此可求手眼矩阵即C->E的坐标系转换矩阵。 |

## 齐次变换矩阵及其使用（~~原理不太懂，~~）

|  |
| --- |
| B ET表示{E}坐标系到{B}坐标系的齐次变换矩阵。及E在基坐标系B下的位姿[23-10-31]。  相机坐标中的一个点转换至基坐标系则有：  PB = B ET × E CT × PC (左乘才表示从下到上，从旧到新，从L->E)  **首先要明确的是齐次变换矩阵是干嘛的，他是将空间中某一坐标系中的向量经过平移旋转至另一坐标系，那么在此基础上就不可能出现P(4\*1) × T(4\*4)的情况。**  ~~那么就有~~  ~~P~~~~L~~ ~~= P~~~~B~~ ~~× B ET × E LT~~~~(右乘表示从上到下，从新到旧，从E->L)~~  ~~即~~  ~~P~~~~E~~ ~~= B ET × P~~~~E~~ ~~× B ET~~  **~~不存在一个点右乘齐次变换矩阵，4x1去哪乘4x4去~~** |

## 三维空间中的向量 – 旋转矩阵

|  |  |
| --- | --- |
| 首先，旋转矩阵是正交矩阵。其次旋转矩阵是坐标系转换中新坐标系的基向量。更准确的，旋转矩阵是用原坐标系表示的新坐标系的基向量。证明见论文7[23-11-3]。  在三维空间中，两个向量的叉乘(外积)是一个垂直于这两个向量组成平面的向量（法向量），向量大小为两个向量围城平行四边形的面积。  那么在旋转矩阵中，任意两个向量的叉乘等于另一向量。  向量点乘a·b的几何意义是a在b上的投影acosθ，数值意义表示a和b的相似度(方向方面，数值越大方向越相似。)  [奇技淫巧系列：向量叉乘 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/148780358) 叉乘计算：   |  | | --- | | [tempDocuments\向量叉乘.txt](tempDocuments/向量叉乘.txt) | |

## 旋转向量、欧拉角、罗德里格斯参数(老迷糊了)

|  |
| --- |
| 旋转矩阵是一个3×3的矩阵，它可以描述一个向量在三维空间中的旋转状态。具体来说，旋转矩阵的每个元素都代表了向量在旋转后在新坐标系中的投影。旋转矩阵的构造方式有多种，其中最常见的是使用三个互相垂直的单位向量来构造旋转矩阵。这三个单位向量通常被称为旋转矩阵的列向量，它们描述了向量在新坐标系中的方向。通过旋转矩阵，您可以方便地描述向量在三维空间中的旋转状态。 |
| 以下是一个绕x、y、z轴依次旋转45度的旋转矩阵：  R = [0.7071 0.7071 0;  -0.5 0.5 0.7071;  0.5 -0.5 0.7071]  将这个旋转矩阵转换成罗德里格斯参数的形式，我们可以使用以下的公式：  theta = acos((trace(R) - 1) / 2)  k = [R(2,3) - R(3,2); R(3,1) - R(1,3); R(1,2) - R(2,1)] / (2 \* sin(theta))  其中，`theta`是旋转角度，`k`是旋转轴向量，trace是矩阵的秩。将上面的旋转矩阵代入公式，可以得到：  theta = 1.9106  k = [0.4082; 0.4082; 0.8165]  因此，这个旋转矩阵可以表示为罗德里格斯参数`[0.4082; 0.4082; 0.8165]`，其中旋转轴为 `[1; 1; 2]`，旋转角度为109.47度。  将旋转矩阵转换成欧拉角的形式，我们可以使用以下的公式：  phi = atan2(R(3,2), R(3,3))  theta = atan2(-R(3,1), sqrt(R(3,2)^2 + R(3,3)^2))  psi = atan2(R(2,1), R(1,1))  其中，`phi`、`theta`和`psi`分别是绕z、y、x轴旋转的欧拉角。将上面的旋转矩阵代入公式，可以得到：  phi = 0.7854  theta = -0.7854  psi = 0.7854  因此，这个旋转矩阵可以表示为欧拉角`[0.7854; -0.7854; 0.7854]`，其中旋转顺序为Z-Y-X。 |
| R(θ)v = v cosθ + (k × v) sinθ + k(k · v)(1 - cosθ)罗德里格斯是描述一个向量(v)绕一个轴(k)旋转一定角度(θ)  这个公式是罗德里格斯公式的一种形式，它描述了向量绕着一个轴旋转一定的角度后的旋转结果。具体来说，这个公式可以将旋转向量v绕着旋转轴k旋转一个角度θ，得到旋转后的向量R(θ)v。其中，cosθ和sinθ分别代表旋转角度的余弦值和正弦值，k × v表示向量k和v的叉积，k · v表示向量k和v的点积，1 - cosθ表示旋转角度的余弦值。这个公式是罗德里格斯公式的一种形式，用于计算向量的旋转后的结果。 |
|  |

## TCP标定和外部工具坐标系建立

|  |
| --- |
|  |
|  |
|  |

## 相机成像

|  |
| --- |
| [UC3D线激光轮廓传感器工作原理及参数说明 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/587449990)  [工业相机（3D）主要参数详述\_3d相机速度计算公式图解-CSDN博客](https://blog.csdn.net/baidu_35536188/article/details/110229065)  [3D视觉 之 线激光3D相机 - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/486516636) |
|  |

## 结果误差分析

|  |
| --- |
| 主流的结果误差分析采用的是以第一组数据为真值，计算后续数据较之它的变换作为误差。  但是，开始就觉得这玩意有问题，如果第一组数值变差很大，后续数据与它的偏差很小，最后的误差看似很小，但是没有实际意义啊。所以对于最后的误差分析需要一个真正的定量真值，而不是假设的虚拟真值！！  下图中，在不同方法的情况下，最后得到的误差都在0.24左右，但是两个手眼矩阵之间的误差却很大，在2mm左右，他们与真值可能还存在一些误差。 |

# 1.⭐误差分析(巧妙思想)-机器人激光检测手眼标定误差分析及优化

#### Path：[Papers\over\手眼标定\_23\_10\_9\_⭐机器人激光检测手眼标定误差分析及优化\_刘建春.pdf](Papers/over/手眼标定_23_10_9_⭐机器人激光检测手眼标定误差分析及优化_刘建春.pdf)

#### 收获小结：eye-to-hand（平均精度0.1495）||首先提出手眼标定三大误差来源，针对其中两种主要来源进行分析并优化。

#### 求解手眼矩阵方法：加权最小二乘法（此外文中还提到了SVD）

## 1.1 硬件设备信息

|  |
| --- |
| **IRB ABB 6700工业机器人**  机器人重复定位精度为0.05mm，线性度为0.096mm。机器人重复定位精度为0.05mm表示，当机器人在同一位置执行多次任务时，它们所到达的位置的最大偏差不超过0.05mm。线性度为0.096mm表示，机器人在执行线性运动时，它们所到达的目标点与理论目标点之间的最大偏差不超过0.096mm。  **LJ-G5001 激光轮廓扫描仪（官网没有这个型号）**  **两个标准球**分别为半径15mm，圆度0.0015mm和半径19.05mm，圆度0.002mm。 |

## 1.2 误差来源分析

|  |
| --- |
| 论文将手眼标定误差总结为三个方面，第一是设备带来的误差（相机、机械臂、标定物等），这个误差是固定的客观存在的，不做考虑；第二是在使用标准圆作为标定物时，对于球心拟合的误差；第三是利用手眼关系对式子进行求解时的误差。  **本文在对误差作分析时，使用实现数据并画出图标作为分析依据。** |

## 1.3 手眼关系及误差表达式

|  |
| --- |
| 大部分论文都会把手眼标定关系转换为AX=XB问题，从而去求解手眼矩阵X，但是本文使用的是B CTC OT=B ETE OT，其中B CT表示相机到基底的转换矩阵（即最终的手眼矩阵）记为A，C OT表示通过拟合圆进而求出的球心坐标记为b，B ETE OT前者表示法兰盘到基底的坐标变换，后者表示标定物及球心坐标到法兰盘的坐标转换（**文中提到此项可以根据硬件设备信息求得，也就是说对于眼在手外，标定物的坐标可以作为已知条件**），两项乘积记为c，则有Abi=ci，那么误差Δ=Abi-ci。补充[23-10-26]：本文中的原意应为B CTC OP=B ETE OP，P是坐标，T是转换矩阵，两者差异巨大。C OP很好理解，就是拟合球心在相机坐标系下的坐标；E OP则是根据标定物的硬件尺寸推算出物理球心在法兰盘坐标系下的坐标。然后利用最优化求解误差方程，得出手眼矩阵A=B CT。 |

## 1.4 \*球心拟合误差（需要看一下论文3中线激光结构光相机的成像原理）

|  |
| --- |
| 根据线激光结构光原理，其成像误差受到入射角的影响，文中提到拟合圆半径越小，则激光光斑区域高度差越大，误差越大（此处应该也与入射角有部分关系）。同时，拟合圆所用到圆弧部分，两端和中间点的误差也很大（使用下面将要说到的优化算法来减小误差）。  在这一部分，文中的逻辑是：首先使用最小二乘法对球心坐标进行拟合，然后利用Ab-c求出误差，分析出拟合半径越大，误差越小；然后，对拟合曲线上的点到拟合圆心的距离进行误差分析，得出曲线两端和中间段的误差较大，进而使用改进最小二乘法进行切面圆的拟合。**而在实际操作中，选择RANSAC可能效果就和改进的最小二乘法相似。但是这一部分的思想很有意思。** |

## 1.5 \*手眼矩阵计算误差及改进

|  |
| --- |
|  |

## 1.6 其他

|  |
| --- |
| 文中最后还是用不同的标准球进行验证。 |

# 2.⭐0.0815(高精度)-基于标准球的机器人手眼标定方法改进研究

#### 收获小结：终于了解了误差分析来源，验证了论文1中的优化方案，精度提高了一个数量级，最大误差仅为0.0815mm

#### 求解手眼矩阵方法：四元数

## 2.1 设备信息

|  |
| --- |
| FANUC 机器人公司的Robot LRMate 200iD 型机器人  机器人重复定位精度为0.02mm，线性度 无。  基恩士轮廓测量仪LJ-V7060 |

## 2.2 手眼关系及标定流程

|  |
| --- |
| 此论文中将第一次测量计算得到的球心作为标准去计算后续球心与其的误差。    **始终的ni·d表述不太正确，应该是（-a+b）·d，负方向移动a次，正方向移动b次。补充[23-10-26]：此处误差会很大{x1, n1d, z1}处的位姿和{x1, 0, z1}处的位姿不同，所以应该直接读取半径最大处的球心坐标和机器人位姿。** |
| 此论文中直接表示对最大半径处进行扫描拟合圆可以更大程度上的减小误差。**初次投射就应该将激光线基本对准最大半径处。** |

## 2.3 其他

|  |
| --- |
| 这篇论文最亮眼的地方是最后的精度，同时基本继承了论文1中提到的误差改进方法。 |

# 3.学位论文-基于3D机器视觉的工业机器人跟踪涂胶系统\_陈琳

#### 收获小结：论文整体架构全面，项目所含技术全面。包含机器人运动学、相机成像原理等

# 4. 线激光器的手眼标定方法

#### 收获小结：简单易操作，误差精度在0.01~0.17mm(标准球)，0.5~5mm(空间定点)。使用几何法求解标准球切面拟合圆(圆上任意四点连线的垂直平分线的交点)。

## 4.1 设备信息

|  |
| --- |
| LMI Gocator2340    ABB IRB120  重复精度0.01mm |

## 4.2 其他

|  |
| --- |
| 方法简单，精度还可以，手眼标定入门文章。  介绍了空间顶点和标准圆（球心作为空间定点）的标定方法。 |

# 5. Pointnet

## 5.1 解决的问题

|  |
| --- |
| 可以对点云进行直接处理 – 网络结构简单 – 鲁棒性强 – 可完成多任务 – 复杂度低 |
|  |

# 6. (难)基于3D 标定块的机器人与3D 相机手眼标定研究

#### 收获小结：矩阵运算和公式很多；多利用3D相机z轴精度高的因素；引言部分介绍许多求解手眼矩阵的方法；Tsai方法的过程即所用公式

## 6.1 没有看懂的地方

|  |
| --- |
| 1. 找寻特征点，利用最小二乘法建立并求解平方方程的地方没有看懂。  2. 利用特征点建立标定物坐标系的地方没有看懂，后面求解手眼矩阵的方法也就没看。 |

## 6.2 文章亮点

|  |
| --- |
| 1.文章中提到z轴精度高，误差小的因素，那在之后的方案设计中就需要考虑到这些因素。  2. 文中的引言介绍了许多求解手眼矩阵的方法，回头可以参考一下参考文献。  3. 文中大量使用了矩阵表达式和矩阵运算，以及一些思想方法的推导过程。 |

## 6.3 TSAI的运算过程

|  |
| --- |
|  |

# 7. (好论文~~垃圾论文~~) 面向物料抓取机器人的新型手眼标定方法

#### 收获小结：更加了解空间坐标系的变换

## 7.1 方法(眼在手外)

|  |
| --- |
| 总的来说，本文利用自制的标定板(规则排列的数个圆)，然后使用B OTO CT = B CT直接获得手眼矩阵，方法简单。~~粗暴，但是精度异常可怜，~~首先是B OT，其获取类似于机器人外部工具坐标系建立的方法~~，且没有改进，所以精度很低。~~所以，机器人的TCP标定以及标定物的坐标系建立成为影响精度的主要因素。  首先在标定板上建立标定板坐标系，然后使用 TCP测量标定板上任意四个点在基坐标系(B)中的坐标，根据四个点的坐标建立标定板坐标系的基向量直接作为旋转矩阵(文中并没有提，但是需要进行归一化才可以作为旋转矩阵，自己画图测试~~误差很大。~~没有问题，很新颖，对空间变换又多了一个理解)    之后根据线性插值法，求解了标定物到相机的转换矩阵C OT，文中没有提什么相机，暂时理解为相机外参。  则手眼矩阵 B CT = B OTO CT = B OTC OT-1 |
|  |

## 7.2 误差分析

|  |
| --- |
| 本文误差在毫米级别，精度很差。从手眼矩阵方程式来看，误差主要来源于:   1. B OT使用TCP去触及标定板上的点获得坐标，那么TCP的精度和获取标定板上点坐标的误差就是B OT的误差来源。 2. O CT – 外参矩阵求解的误差 |

## 7.3 改进方案(待完善)

|  |
| --- |
| 1. 标定板上坐标需要多获取几组，同时要保证机器人的多姿态  2. B OT的求解方法(如何从多组点中求出精度最高的坐标系向量) |
|  |

# 8. 正交约束下的机器人线结构光手眼标定

#### 收获小结：使用标定板测量，新的手眼矩阵关系

(多环境验证，平均精度不超过0.15mm，但是论文结尾说系统误差在0.3mm左右)

## 8.1 设备信息(硬件太好)

|  |
| --- |
| 运动机构采用ABB 公司的S7－1200 六轴工业机器人，重复定位精度±0. 01 mm; 线结构光传感器采用北京同舟兴业科技有限公司的GF3－150，测量精度±0. 03 mm; 使用高精度数控中心加工的方形不锈钢平板作为标定靶标，边长为200 mm，直角精度优于±0. 05°。 |

## 8.2 方法(眼在手上)

|  |
| --- |
| 使用带有正交关系的标定物，利用其正交关系进行手眼矩阵的求解。同时使用了新的手眼关系求解方程。B ET x E CT x PC=PB ,然后进行运算整合有Ax=b(L指相机坐标系)：    控制系统在具有正交关系的边上分别取N个点，每两个点可组成一条向量且与另一条边正交，即：    求解可得手眼矩阵。 |

## 8.3 误差验证方法

|  |
| --- |
| 使用现有手眼矩阵，求解各点在机器人基坐标系下的坐标，然后进行直线拟合和平面拟合计算手眼标定误差。 |

## 8.4 误差分析 & 方法迁移

|  |
| --- |
| 手眼关系等式：B ET x E CT x PC=PB 眼在手外则为 B CT x PC=PB  对于眼在手上而言误差主要来源于在标定物上取点和最优化方程的求解。~~而对于眼在手外而言无法直接使用这个方法，因为如果把标定物放在机械臂上，则无法保证采集到的点所在的边是正交关系。但是可以将标定物固定，使用TCP去有正交关系的边上取点，~~眼在手外也可以使用这个方法，因为对于眼在手外而言，并不需要改变机器人的位姿去提高手眼矩阵的精度，其要求的转换关系中的两个坐标系都是固定不动的，受到额外误差的影响很少。使用该方法的话，直接套就好了。只是本文的方法中矩阵A是已知参数，向量x是未知参数，我们的唯一一个矩阵就是我们要求的手眼矩阵。 |

# 9. PointNN\_CVPR2023\_Starting\_From\_Non-Parametric\_Networks\_for\_3D\_Point\_Cloud\_Analysis\_CVPR\_2023\_paper

## 0. 细节不太懂-无参数-可作插件-性能可

## 1. 总结

|  |
| --- |
| 1.1 动机  自从Pointnet开始使用深度学习直接处理点云后，大家都在追求更高的性能的同时，网络的参数和需要的算力都越来约高。   * 1. 做了什么   提出了一种无参数的模块，此模块主要利用FPS、K紧邻、池化以及transformer中的三角变换来实现。  同时，此模块可通过添加简单的线性层来提高性能，也可以作为插件，让已经训练好的网络进一步提升性能。   * 1. 取得的成绩   详见论文 |

## 技术路线

|  |
| --- |
|  |

# 10. ECCV\_2020 PointContrast: Unsupervised Pre-training for 3D Point Cloud Understanding

## 10.1 点云深度学习网路现状

|  |
| --- |
| 1）缺乏大规模和高质量的数据：与2D图像相比，3D数据更难收集，标记成本更高，并且传感设备的多样性可能会带来巨大的领域差距;2）缺乏统一的骨干网架构：与2D视觉相比，ResNets等架构已被证明是成功的预训练和网络调优的骨干网，点云网络架构设计仍在不断发展;3）缺乏一套全面的数据集和用于评估的高级任务。 |

## 10.2 论文主要贡献 – 预训练网络

|  |
| --- |
| 在相同世界坐标中对齐生成的两个视图[使用KDTree 或者八叉树]（x1和x2），即它们在相同的世界坐标系下表示相同的空间位置。确保两个视图之间的点能够有意义地进行对比和匹配。  使用对应关系映射M建立两个视图之间的点的对应关系。具体来说，如果(i, j) ∈ M，则表示在两个视图中的点x1i和x2j是匹配的，即它们在相同的物体表面上。  该预训练任务的框架是为了学习点云的表示，其中点云是三维空间中的一组点。首先，从分布中采样一个点云x，然后生成两个在相同世界坐标中对齐的视图x1和x2。接着，计算这两个视图之间的对应关系映射M，以获取匹配的点对。为了增加任务的难度，引入了随机的刚性几何变换T1和T2，如旋转、平移和缩放，将两个视图进一步变换。最后，通过对两个视图中的点应用对比损失来训练模型，以使匹配的点更加接近，而不匹配的点更加远离。这个方法与2D图像的对比学习相似，但在这里对比学习是在点级别进行的，目的是学习点云的紧凑表示，特别是用于几何局部描述符的度量学习。 |

### 10.2.1 损失函数

|  |
| --- |
| 第二个名为PointInfoNCE参考了2D处理中的InfoNCE(对比学习) |

### 10.2.2 基于U-Net的SRU-Net网络

|  |
| --- |
|  |

# 11. CVPR\_2020\_GraphTER\_Unsupervised\_Learning\_of\_Graph\_Transformation\_Equivariant\_Representations\_via\_Auto-Encoding\_CVPR\_2020\_paper

## 0. (无监督)与论文10异同

|  |
| --- |
| 同为无监督网路处理点云，论文10中表达的思想是进行预训练 – 通过两个不同的视点进行网络训练  在本文中，有异曲同工之处，论文10可以看作是对整点云进行(旋转)变换，而在本文中也是对点云进行仿射变换，但是是在点级的基础上，且不仅限于旋转，包含平移、旋转和裁剪。 |

## 算法流程

|  |
| --- |
| 算法的流程如下： 步骤1：给定一个包含N个节点的图形，将其表示为图信号X = {x1, x2, …, xN}。然后，在训练的每个迭代中，从图中随机选择一个节点子集S。节点子集可以通过全局采样或局部采样的方式获取。局部采样是在图中的局部节点集合进行采样，而全局采样是在整个图中进行随机采样。这样做的目的是将全局和局部的图结构性质以不同尺度进行建模，并提高计算效率[4a]。 步骤2：在子集S的每个样本xi上绘制一个节点变换ti，这可以是各向同性或各向异性的。对应地，与变换图相关的图A˜也会在应用样本节点变换之后在A的基础上进行等变变换。节点变换t的选择会根据节点的特征和图的连接情况来确定[5a]。 步骤3：构造一个基于k最近邻的图，以利用节点之间的连接。这个图的邻接矩阵A在应用了特定节点变换之后会发生变化。然后，使用多层常规边卷积将其堆叠起来，形成最终的编码器。边卷积通过边权重˜ai,j将节点的特征与相邻节点的特征聚合起来。由于底层图的边信息与节点的变换相对应，边卷积可以从原始和变换后的边信息中提取高级特征[5b][5c]。 步骤4：使用一个自动编码器（auto-encoder）学习图信号的表示和节点变换的解码。编码器将节点特征映射到连续表示，解码器从这个连续表示中重建原始和经过变换的图的特征。节点变换t通过解码器来估计，然后反向传播损失来更新编码器和解码器的参数[4b][4c][5d]。 步骤5：通过最小化损失函数来端到端地训练整个网络[4d]。损失函数的期望是在每次训练的迭代中，通过对采样子集S上的节点变换进行评估而计算的[4d]。 |

12. CVPR\_2022\_CrossPoint\_Self-Supervised\_Cross-odal\_Contrastive\_Learning\_for\_3D\_Point\_Cloud\_Understanding

## 12.1 文献阅读

|  |
| --- |
| [浅读CVPR2022-CrossPoint - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/482785806) |
| [再读CVPR2022-CrossPoint（附代码） - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/484988114) |