

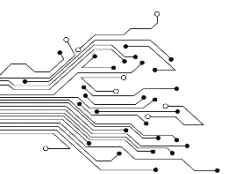


点云模型

@八斗学院--王小天(Michael)

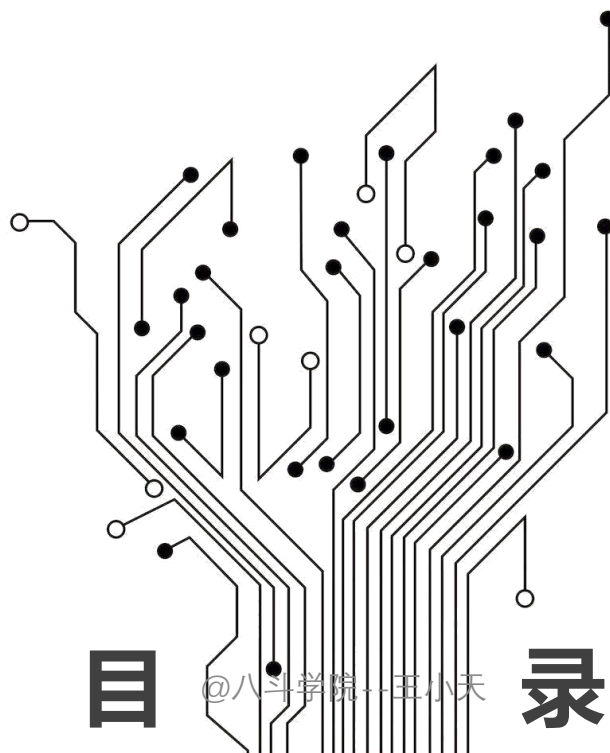
2021/7/18

@八斗学院--王小天

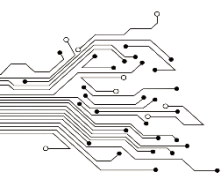


---八斗人工智能，盗版必究---

1. 点云模型
2. Spin image
3. 拓展-三维重建



@八斗学院--王小天



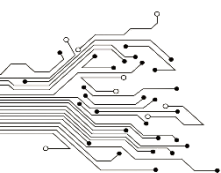
三维图像是一种特殊的信息表达形式，其特征是表达的空间中三个维度的数据。和二维图像相比，三维图像借助第三个维度的信息，可以实现天然的物体和背景解耦。

对于视觉测量来说，物体的二维信息往往随射影方式而变化，但其三维特征对不同测量方式具有更好的统一性。

与相片不同，三维图像是对一类信息的统称，信息还需要有具体的表现形式，其表现形式包括：深度图（以灰度表达物体与相机的距离），几何模型（由CAD软件建立），点云模型（所有逆向工程设备都将物体采样成点云）。

点云数据是最为常见也是最基础的三维模型。

点云：扫描资料以点的形式记录，每一个点包含有三维坐标，有些可能含有颜色信息（RGB）或反射强度信息（Intensity）。



点云的概念：点云是在同一空间参考系下表达目标空间分布和目标表面特性的海量点集合，在获取物体表面每个采样点的空间坐标后，得到的是点的集合，称之为“点云”（Point Cloud）。

点云的内容：根据激光测量原理得到的点云，包括三维坐标（XYZ）和激光反射强度（Intensity），强度信息与目标的表面材质、粗糙度、入射角方向，以及仪器的发射能量，激光波长有关。

根据摄影测量原理得到的点云，包括三维坐标（XYZ）和颜色信息（RGB）。

结合激光测量和摄影测量原理得到点云，包括三维坐标（XYZ）、激光反射强度（Intensity）和颜色信息（RGB）。



点云处理的三个层次

低层次处理方法:

- ①滤波方法：双边滤波、高斯滤波、条件滤波、直通滤波、随机采样一致性滤波。
- ②关键点：ISS3D、Harris3D、NARF, SIFT3D

中层次处理方法:

- ①特征描述：法线和曲率的计算、特征值分析、SHOT、PFH、FPFH、3D Shape Context、Spin Image
- ②分割与分类：
 - 分割：区域生长、Ransac线面提取、全局优化平面提取
 - K-Means、Normalize Cut (Context based)
 - 3D Hough Transform(线、面提取)、连通分析
 - 分类：基于点的分类，基于分割的分类，基于深度学习的分类 (PointNet, OctNet)



高层次处理方法：

①配准：点云配准分为粗配准（Coarse Registration）和精配准（Fine Registration）两个阶段。

精配准的目的是在粗配准的基础上让点云之间的空间位置差别最小化。

粗配准是指在点云相对位姿完全未知的情况下对点云进行配准，可以为精配准提供良好的初始值。

基于穷举搜索的配准算法：遍历整个变换空间以选取使误差函数最小化的变换关系或者列举出使最多点对满足的变换关系。如RANSAC配准算法、四点一致集配准算法（4-Point Congruent Set, 4PCS）、Super4PCS算法等……

基于特征匹配的配准算法：通过被测物体本身所具备的形态特性构建点云间的匹配对应，然后采用相关算法对变换关系进行估计。如基于点FPFH特征的SAC-IA、FGR等算法、基于点SHOT特征的AO算法以及基于线特征的ICL等…

②SLAM图优化

Ceres（Google的最小二乘优化库，很强大）， g2o、LUM、ELCH、Toro、SPA

SLAM方法：ICP、MBICP、IDC、likelihood Field、NDT

③三维重建

泊松重建、Delaunay triangulations、表面重建，人体重建，建筑物重建，树木重建。

实时重建：重建植被或者农作物的4D（3D+时间）生长态势；人体姿势识别；表情识别；

④点云数据管理：点云压缩，点云索引（KD、Octree），点云LOD（金字塔），海量点云的渲染



spin image

---八斗人工智能，盗版必究---

Spin image是基于点云空间分布的最经典的特征描述方法。

Spin image的思想是将一定区域的点云分布转换成二维的spin image，然后对场景和模型的spin images进行相似性度量



生成spin image的步骤

1. 定义一个Oriented point
2. 以Oriented point为轴生成一个圆柱坐标系
3. 定义Spin image的参数，Spin image是一个具有一定大小（行数列数）、分辨率（二维网格大小）的二维图像（或者说网格）。
4. 将圆柱体内的三维坐标投影到二维Spin image，这一过程可以理解为一个Spin image绕着法向量 n 旋转360度，Spin image扫到的三维空间的点会落到Spin image的网格中。
5. 根据spin image中的每个网格中落入的点不同，计算每个网格的强度 I



生成spin image的步骤

---八斗人工智能，盗版必究---

大写P-----三维网格某顶点p的切面

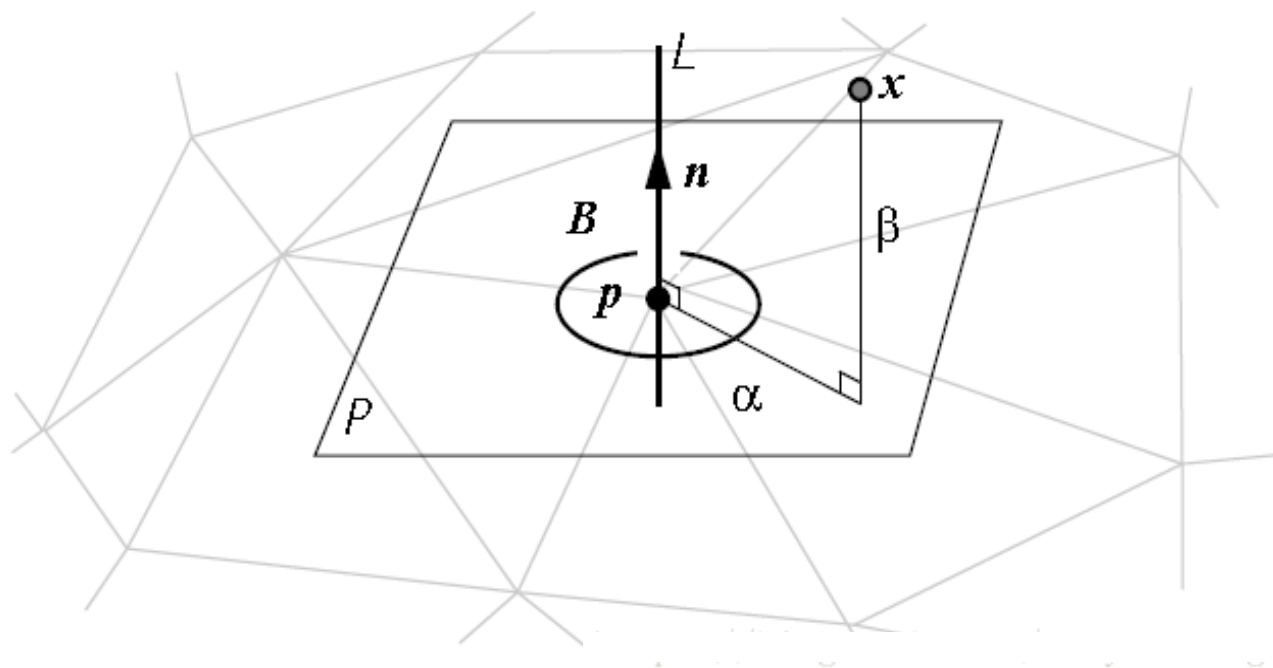
n-----p点单位法向量

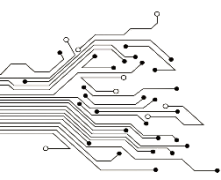
x-----p附近的三维网格上的另一个顶点

α -----x点在P上投影与p的距离

β -----x点与P点的垂直距离

其中p和n定义为一个定向点(Oriented point)





生成spin image的步骤

1. 定义一个Oriented point
2. 以Oriented point为轴生成一个圆柱坐标系
3. 定义Spin image的参数，Spin image是一个具有一定大小（行数列数）、分辨率（二维网格大小）的二维图像（或者说网格）。
4. 将圆柱体内的三维坐标投影到二维Spin image，这一过程可以理解为一个Spin image绕着法向量 n 旋转360度，Spin image扫到的三维空间的点会落到Spin image的网格中。
5. 根据spin image中的每个网格中落入的点不同，计算每个网格的强度 I ,



spin image的三个关键参数

a.分辨率，即二维网格的也就是像素的实际尺寸，使用和三维网格相近的尺寸比较合适，因此通常是取三维网格所有边的平均值来作为spin image的每个网格尺寸，通常会把网格的长和宽定义成相等，即边长。边长的计算公式（e为三维网格模型中的一条边，N为三维网格模型中的边的总数。）：

$$r = \frac{1}{N} \sum_{i=1}^N |e_i|$$

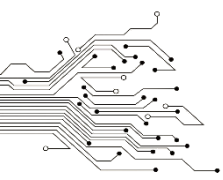
b.大小，即spin image的行数和列数，两者一般也相等。可以参考的大小10x10或20x20等。

c.support angle，即法向量夹角的大小限制。空间中顶点的法向量与创建圆柱坐标系所选点法向量之间的夹角。



生成spin image的步骤

1. 定义一个Oriented point
2. 以Oriented point为轴生成一个圆柱坐标系
3. 定义Spin image的参数，Spin image是一个具有一定大小（行数列数）、分辨率（二维网格大小）的二维图像（或者说网格）。
4. 将圆柱体内的三维坐标投影到二维Spin image，这一过程可以理解为一个Spin image绕着法向量 n 旋转360度，Spin image扫到的三维空间的点会落到Spin image的网格中。
5. 根据spin image中的每个网格中落入的点不同，计算每个网格的强度 I

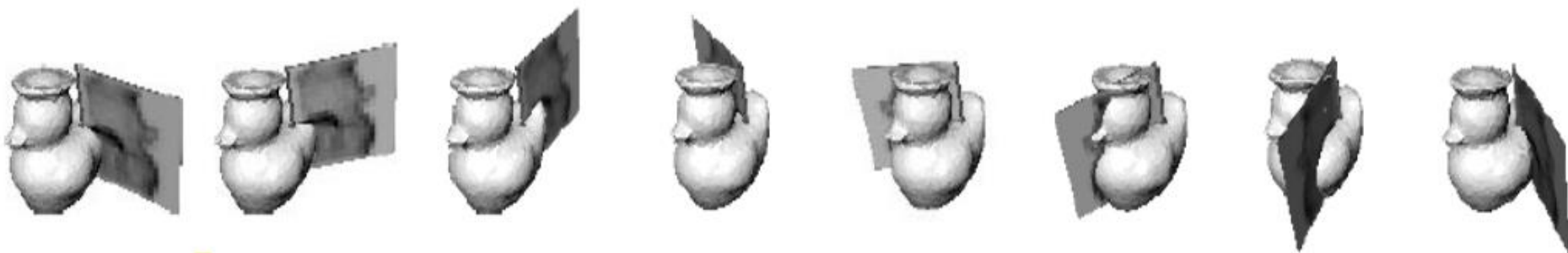


生成spin image的步骤

---八斗人工智能，盗版必究---

4. 将圆柱体内的三维坐标投影到二维Spin image，这一过程可以理解为一个Spin image绕着法向量 n 旋转360度，Spin image扫到的三维空间的点会落到Spin image的网格中。

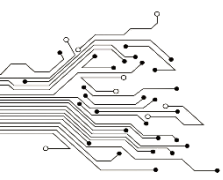
从三维空间投影到spin-image坐标：





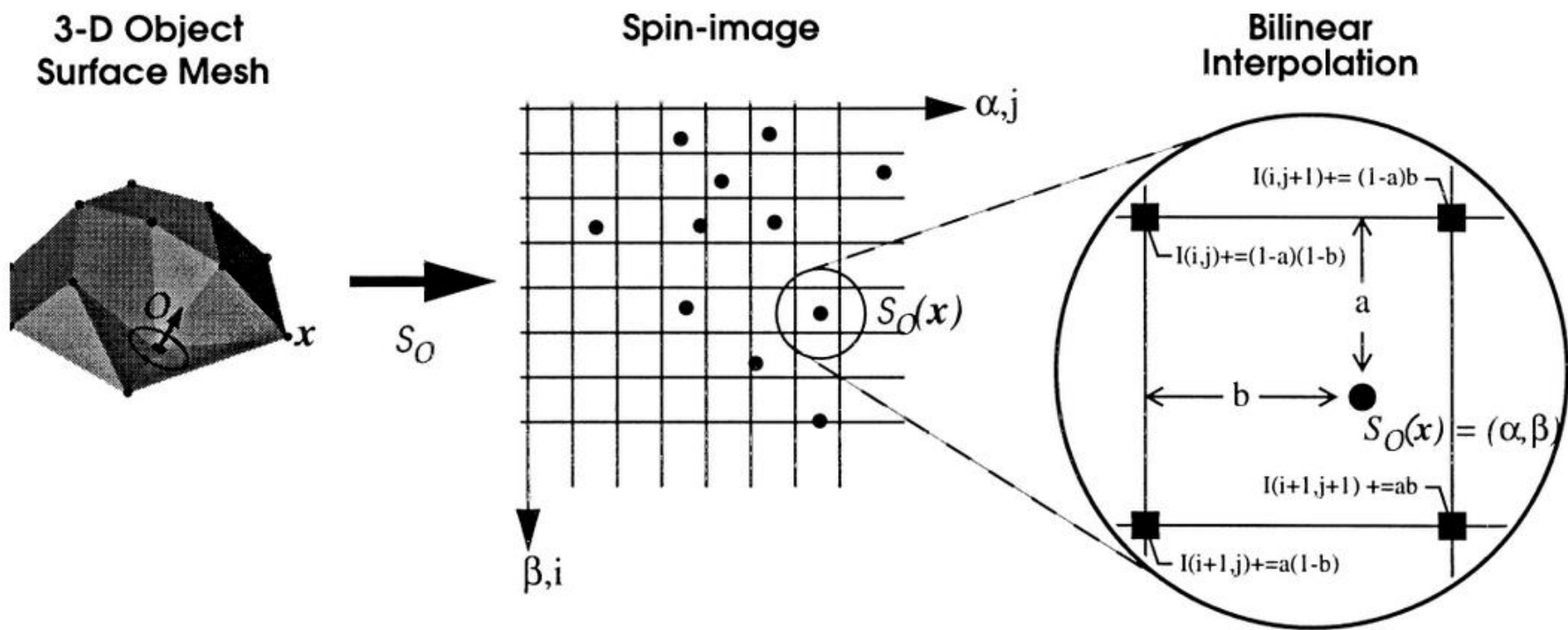
生成spin image的步骤

1. 定义一个Oriented point
2. 以Oriented point为轴生成一个圆柱坐标系
3. 定义Spin image的参数，Spin image是一个具有一定大小（行数列数）、分辨率（二维网格大小）的二维图像（或者说网格）。
4. 将圆柱体内的三维坐标投影到二维Spin image，这一过程可以理解为一个Spin image绕着法向量 n 旋转360度，Spin image扫到的三维空间的点会落到Spin image的网格中。
5. 根据spin image中的每个网格中落入的点不同，计算每个网格的强度 I ,

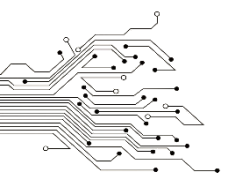


生成spin image的步骤

5.根据spin image中的每个网格中落入的点不同，计算每个网格的强度 I ，显示spin image时以每个网格（也就是像素） I 为依据。最直接的方法是直接计算每个网格中落入的点的个数，然而为了降低对位置的敏感度、降低噪音影响、增加稳定性，Johnson论文中用双线性插值的方法将一个点分布到4个像素中。



当一个点落入网格 (i, j) 中时会被双线性插值分散到 (i, j) 、 $(i, j+1)$ 、 $(i+1, j)$ 、 $(i+1, j+1)$ 四个网格中。



生成spin image的步骤

---八斗人工智能，盗版必究---

拓展：binsize b, 图像尺寸W

计算落在那个bin中的
的计算公式

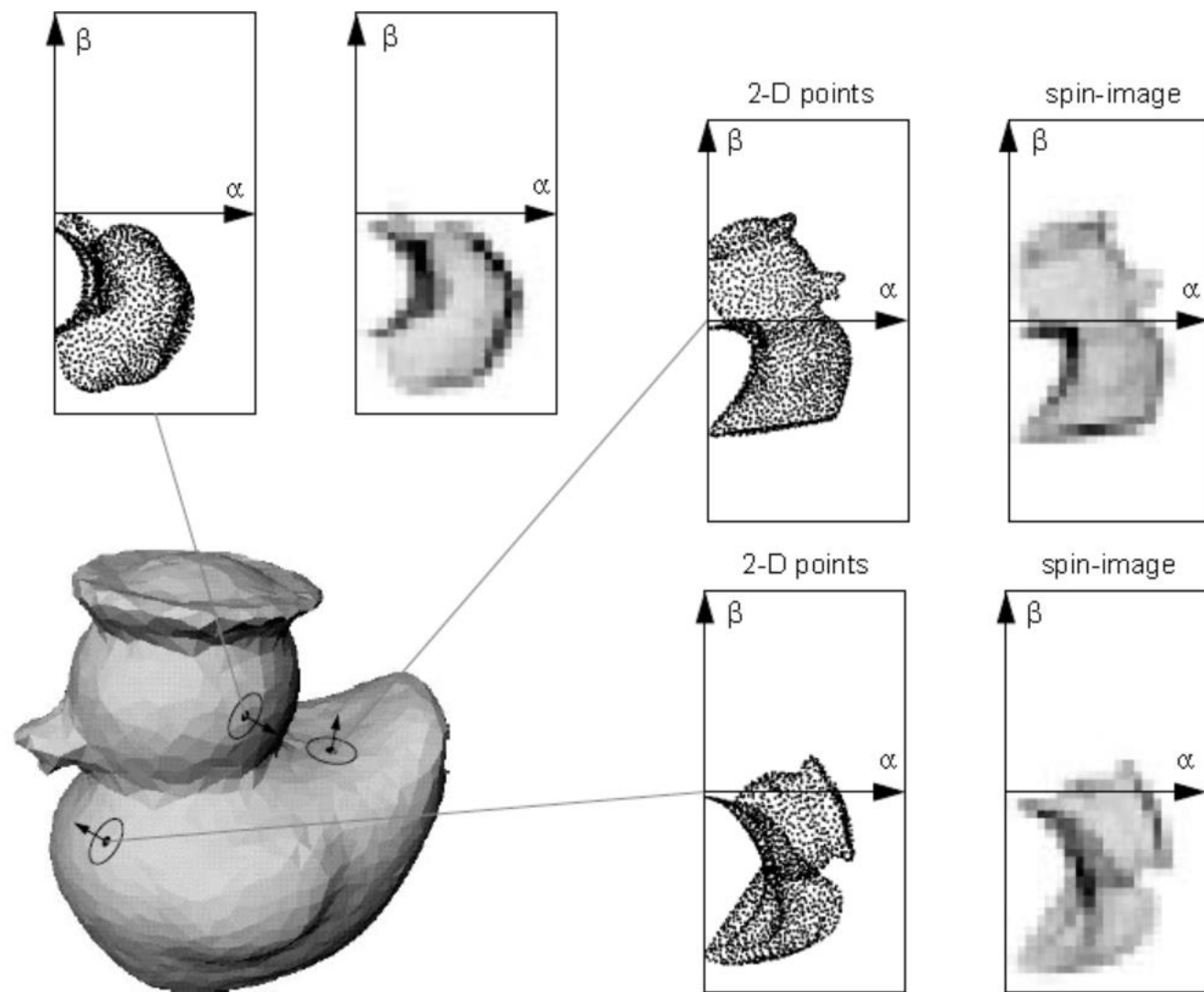


$$i = \left\lfloor \frac{\frac{W}{2} - \beta}{b} \right\rfloor \quad j = \left\lfloor \frac{\alpha}{b} \right\rfloor$$



生成spin image的步骤

这样就获得了spin image:

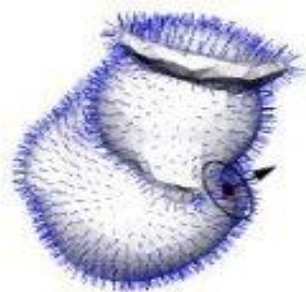


---八斗人工智能，盗版必究---

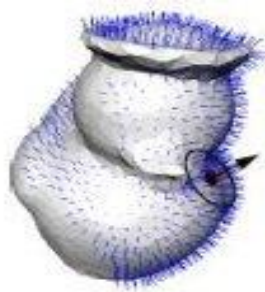


spin image的三个关键参数

c.support angle, 即法向量夹角的大小限制。空间中顶点的法向量与创建圆柱坐标系所选点法向量之间的夹角。



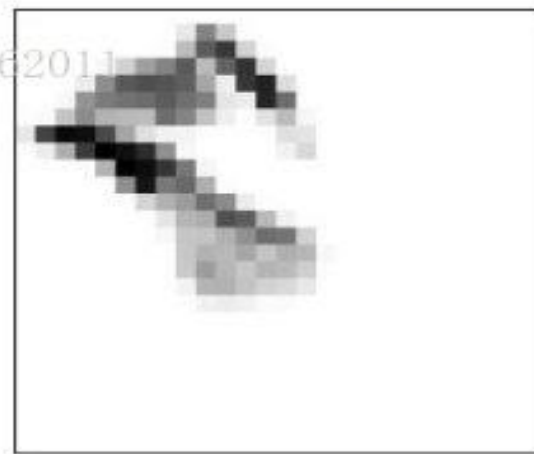
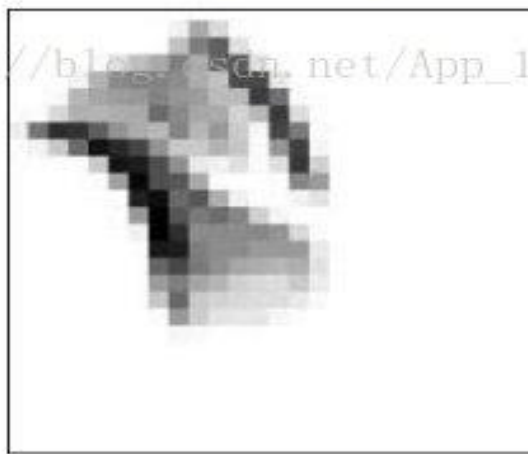
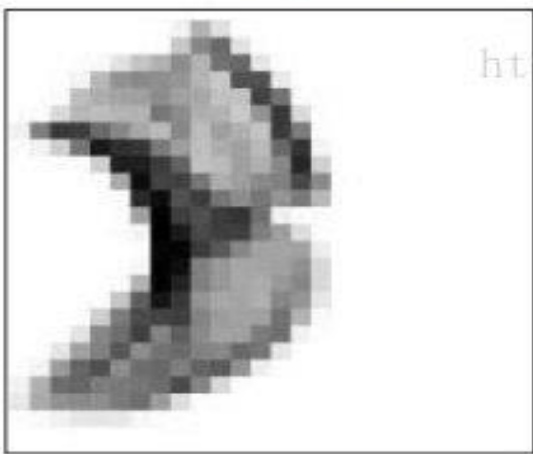
180° support angle



90° support angle



60° support angle



可以看出，对角度限制以后，那些相当于切面的“凹点（大于90°）”被剔除，保留了主要信息，降低了后续的计算量。

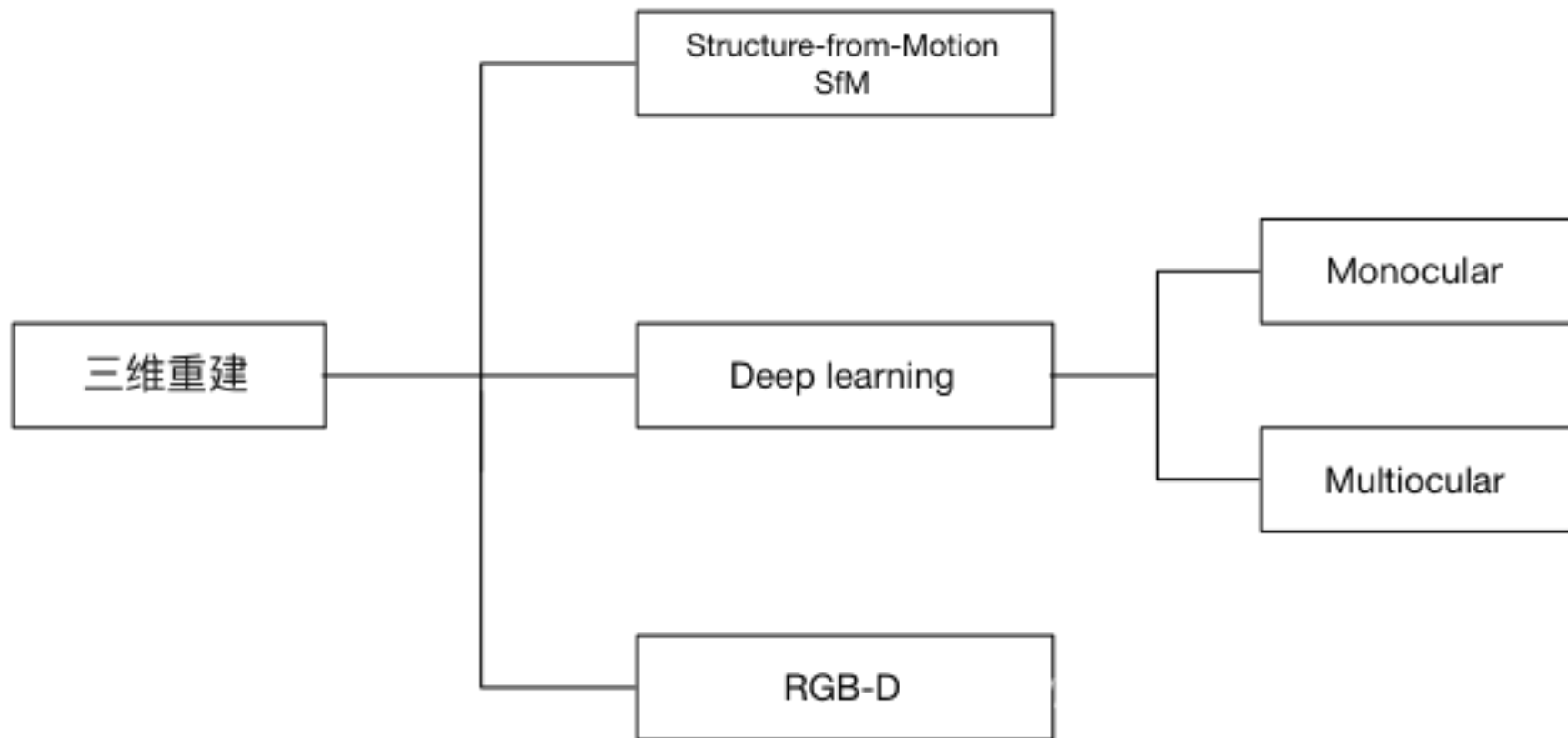
一般角度限制范围为60°~90°之间。



拓展-三维重建

---八斗人工智能，盗版必究---

三维重建包含三个方面，基于SfM的运动恢复结构，基于Deep learning的深度估计和结构重建，以及基于RGB-D深度摄像头的三维重建。





拓展-三维重建

SfM (Structure From Motion)，主要基于多视觉几何原理，用于从运动中实现3D重建，也就是从无时间序列的2D图像中推算三维信息，是计算机视觉学科的重要分支。广泛应用于AR/VR，自动驾驶等领域。虽然SfM主要基于多视觉几何原理，随着CNN的在二维图像的积累，很多基于CNN的2D深度估计取得一定效果，用CNN探索三维重建也是不断深入的课题。

深度学习方法呈现上升趋势，但是传统基于多视几何方法热情不减，实际应用以多视几何为主，深度学习的方法离实用还有一定的距离。



拓展-三维重建

---八斗人工智能，盗版必究---





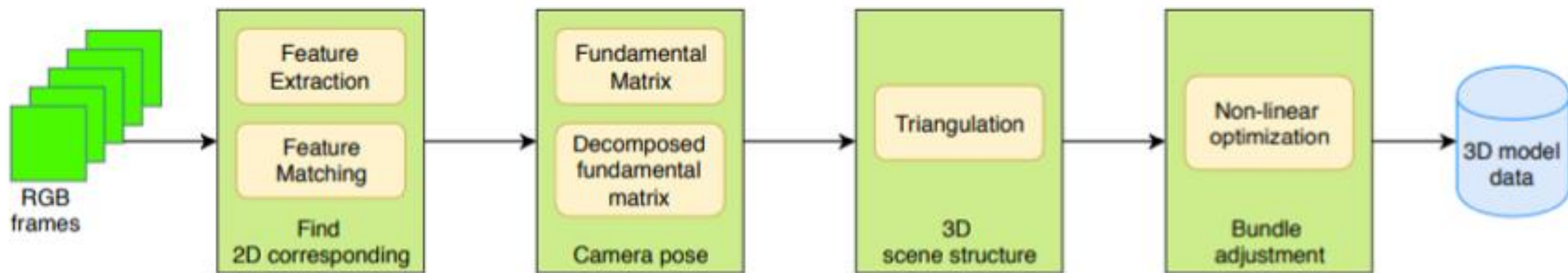
拓展-Sfm与三维重建

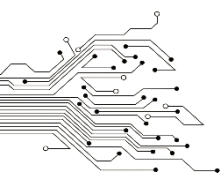
从二维图像中恢复三维场景结构是计算机视觉的基本任务，广泛应用于3D导航、3D打印、虚拟游戏等。

Structure from Motion (SfM) 是一个估计相机参数及三维点位置的问题。

一个基本的SfM pipeline可以描述为:

1. 对每张2维图片检测特征点 (feature point)，对每对图片中的特征点进行匹配，只保留满足几何约束的匹配。
2. 执行一个迭代式的、鲁棒的SfM方法来恢复摄像机的内参 (intrinsic parameter) 和外参(extrinsic parameter)。
3. 并由三角化得到三维点坐标。



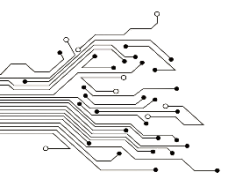


拓展-Sfm与三维重建

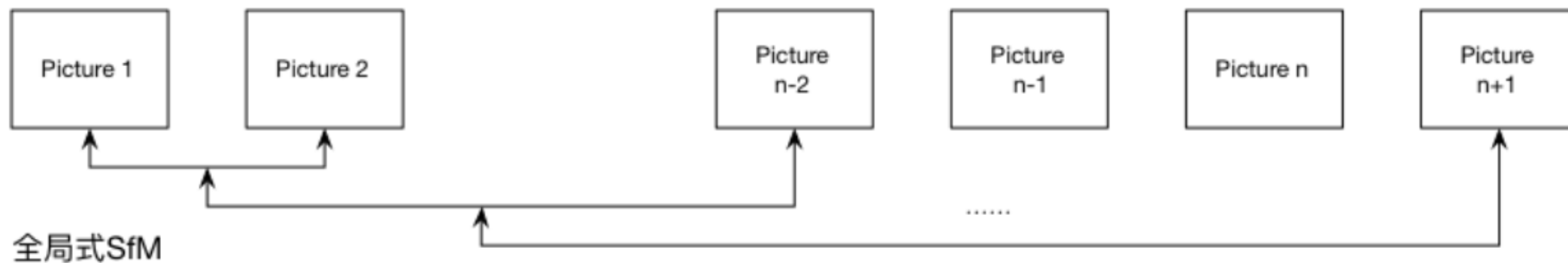
---八斗人工智能，盗版必究---

根据SfM过程中图像添加顺序的拓扑结构，SfM方法可以分为：

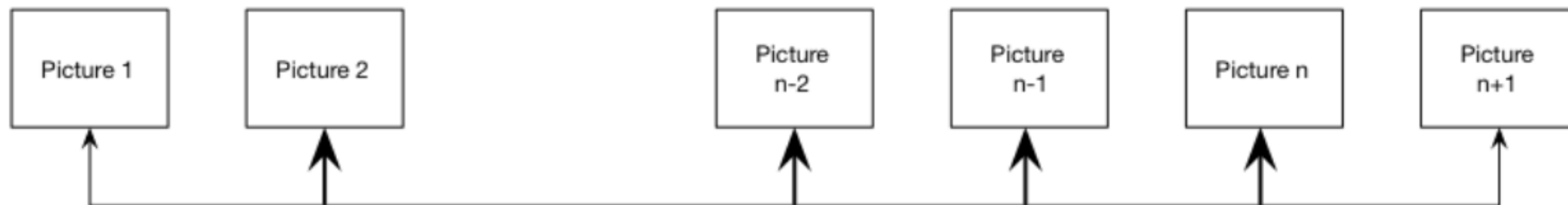
- 增量式 (incremental/sequential SfM)
- 全局式 (global SfM)
- 混合式 (hybrid SfM)
- 层次式 (hierarchical SfM)
- 基于语义的SfM(Semantic SfM)
- 基于Deep learning的SfM



增量式SfM



全局式SfM

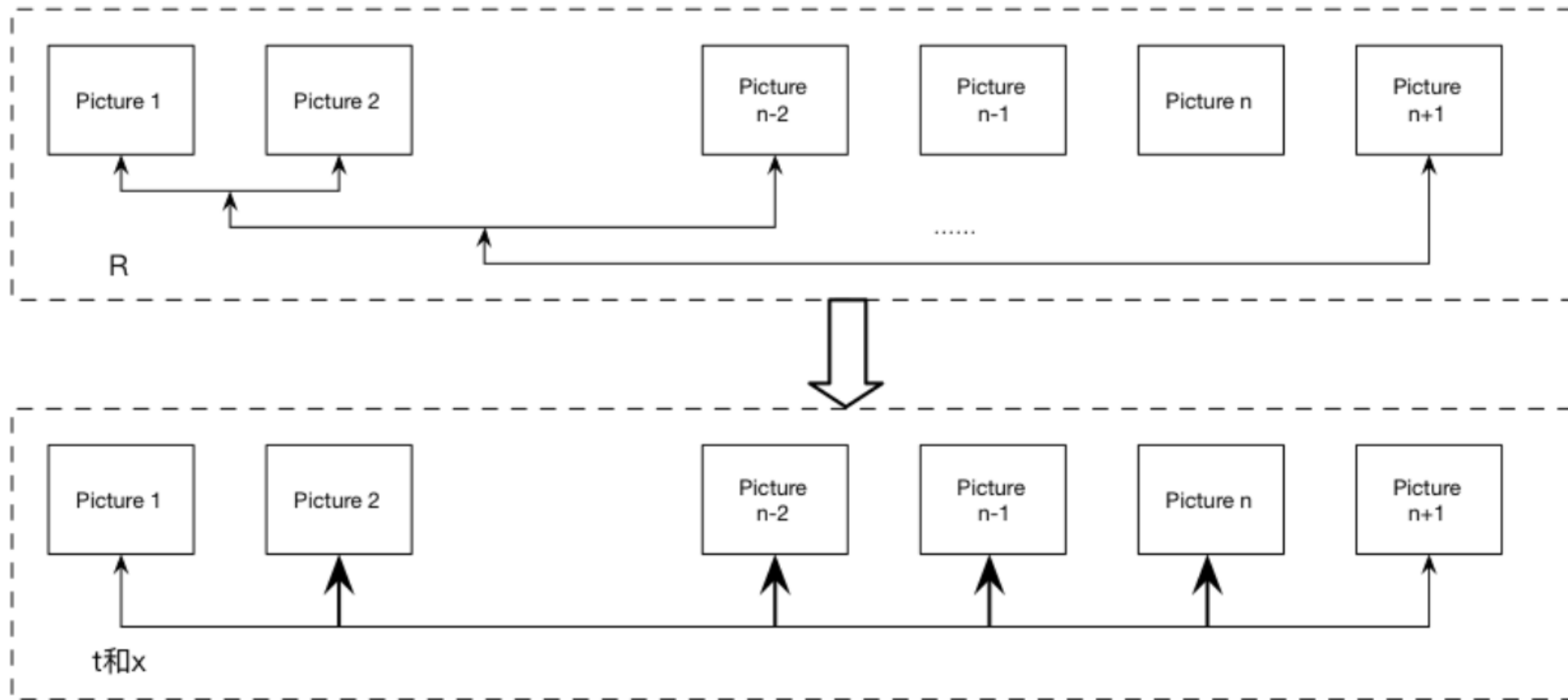




拓展-Sfm与三维重建

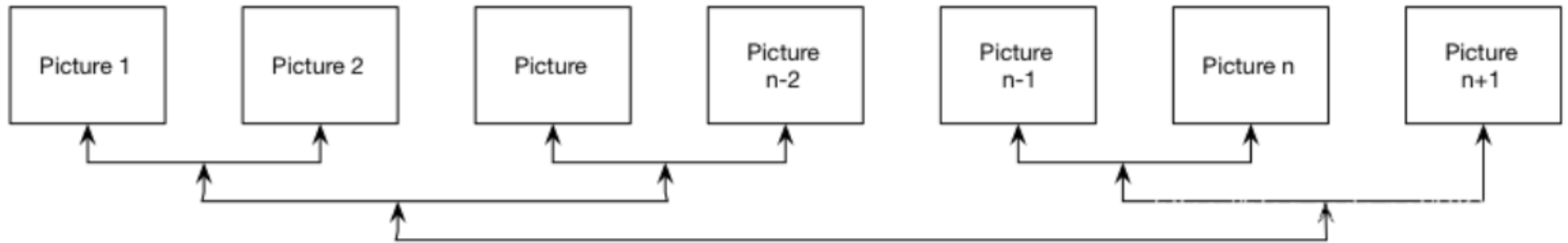
---八斗人工智能，盗版必究---

混合式SfM





层次式SfM



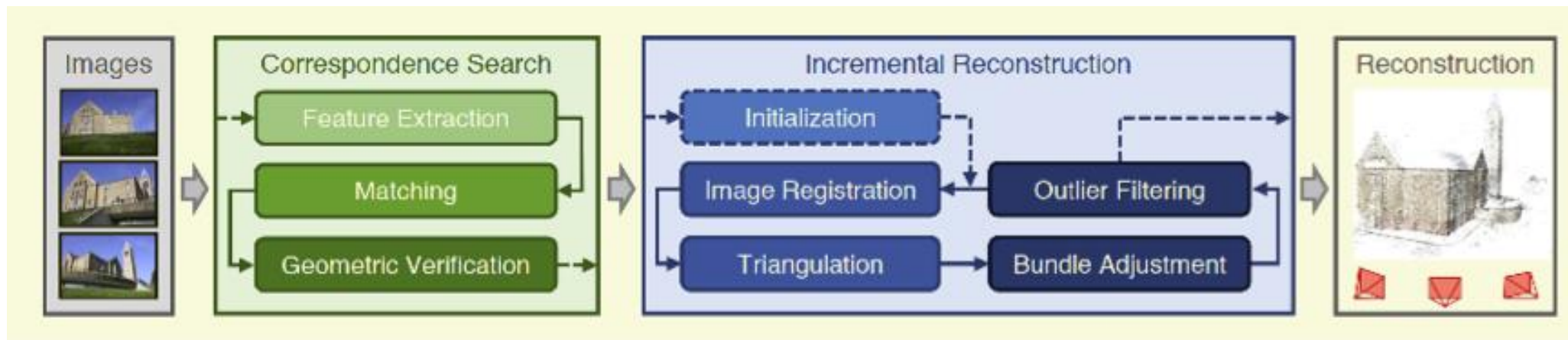


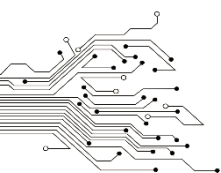
拓展-增量式Sfm

增量式SfM首先使用SIFT特征检测器提取特征点并计算特征点对应的描述子（descriptor），然后使用ANN（approximate nearest neighbor）方法进行匹配，低于某个匹配数阈值的匹配对将会被移除。对于保留下来的匹配对，使用RANSAC（Random Sample Consensus）来估计基本矩阵（fundamental matrix），在估计基本矩阵时被判定为外点（outlier）的匹配被看作是错误的匹配而被移除。对于满足以上几何约束的匹配对，将被合并为tracks。然后通过incremental方式的SfM方法来恢复场景结构。首先需要选择一对好的初始匹配对，一对好的初始匹配对应该满足：

- （1）足够多的匹配点；
- （2）宽基线。之后增量式地增加摄像机，估计摄像机的内外参并由三角化得到三维点坐标，然后使用Bundle Adjustment进行优化。

增量式SfM从无序图像集合计算三维重建的常用方法。增量式SfM可分为几个阶段：图像特征提取、特征匹配、几何约束、重建初始化、图像注册、三角化、outlier过滤、Bundle adjustment等步骤。





拓展-增量式Sfm

---八斗人工智能，盗版必究---

增量式SfM优势：

1. 系统对于特征匹配以及外极几何关系的外点比较鲁棒，重建场景精度高；
2. 标定过程中通过RANSAC不断过滤外点；
3. 捆绑调整不断地优化场景结构。

增量式SfM缺点：

1. 对初始图像的选择及摄像机的添加顺序敏感；
2. 场景漂移，大场景重建时的累计误差。
3. 效率不足，反复的捆绑调整需要大量的计算时间。



拓展-全局式Sfm

全局式：估计所有摄像机的旋转矩阵和位置并三角化初始场景点。

优势：

1. 将误差均匀分布在外极几何图上，没有累计误差。
2. 不需要考虑初始图像和图像添加顺序的问题。
3. 仅执行一次捆绑调整，重建效率高。

缺点：

1. 鲁棒性不足，旋转矩阵求解时L1范数对外点相对鲁棒，而摄像机位置求解时相对平移关系对匹配外点比较敏感。
2. 场景完整性，过滤外极几何边，可能丢失部分图像。



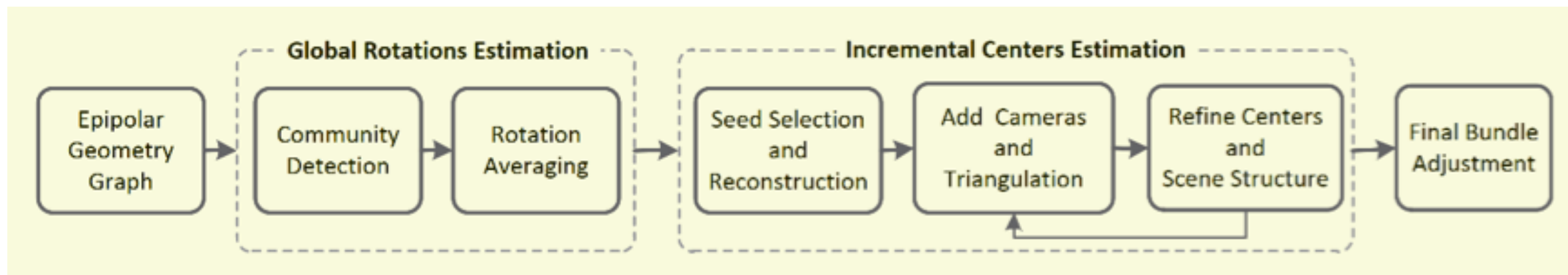
拓展-混合式Sfm

---八斗人工智能，盗版必究---

混合式SfM在一定程度上综合了增量式和全局式SfM各自的优点。

混合式SfM的整个pipeline可以概括为：

全局估计摄像机旋转矩阵，增量估计摄像机位置，三角化初始场景点。





拓展--OpenMVG

openMVG (Open Multiple View Geometry): 开源多视角立体几何库，这是一个cv界处理多视角立体几何的著名开源库，信奉“简单，可维护”，提供了一套强大的接口，每个模块都被测试过，尽力提供一致可靠的体验。

openMVG实现以下典型应用：

1. 解决多视角立体几何的精准匹配问题；
2. 提供一系列SfM需要用到的特征提取和匹配方法；
3. 完整的SfM工具链（校正，参估，重建，表面处理等）；
4. openMVG尽力提供可读性性强的代码，方便开发者二次开发，核心功能是尽量精简的，所以你可能需要其它库来完善你的系统。



拓展--OpenMVG

---八斗人工智能，盗版必究---

github地址: <https://github.com/openMVG/openMVG>

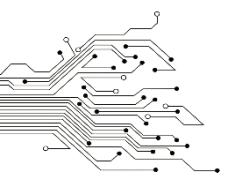
- 从git克隆openMVG到本地（支持windows /linux/mac环境），安装必要的依赖库（缺少依赖库时cmake或make会报错，也可根据提示安装）。

```
$ git clone --recursive https://github.com/openMVG/openMVG.git
$ mkdir openMVG_Build
$ cd openMVG_Build
$ cmake -DCMAKE_BUILD_TYPE=RELEASE -DOpenMVG_BUILD_TESTS=ON -
DOpenMVG_BUILD_EXAMPLES=ON . ../openMVG/src/
$ make
$ make test
```



拓展--OpenMVG

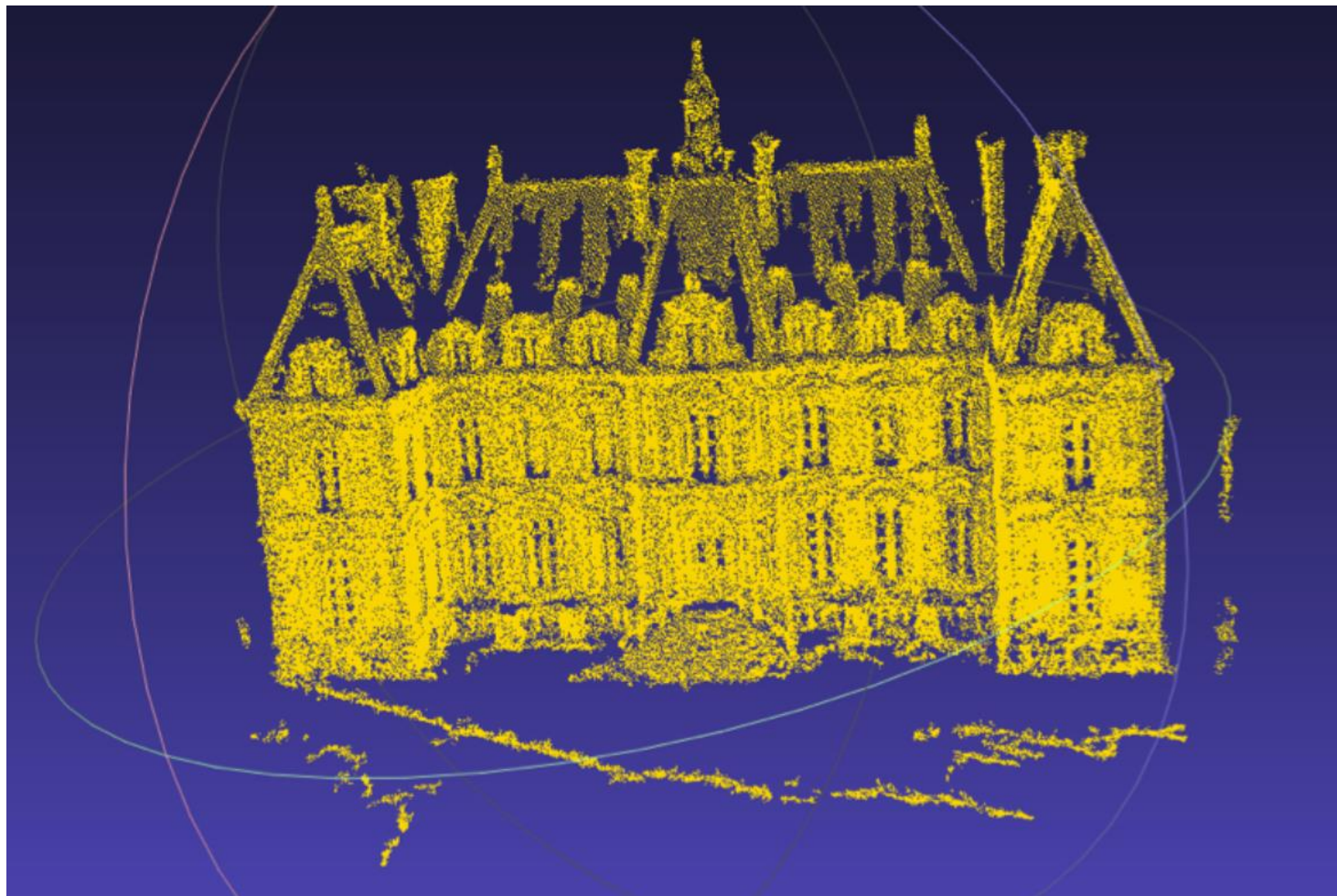
- CMVS-PMVS (a modified version) : 将运动结构 (SfM) 软件的输出作为输入, 然后将输入图像分解成一组可管理大小的图像簇。MVS软件可以用来独立和并行地处理每个簇, 其中来自所有簇的重建不错过任何细节。
- Github地址: <https://github.com/pmoulon/CMVS-PMVS>
- CMVS-PMVS编译也比较简单, 切换到克隆源码目录 (build/main文件夹中生成cmvs、genOption、pmvs2三个可执行文件。) :
 - \$ mkdir build && cd build
 - \$ cmake ..
 - \$ make
- cd到openMVG_Build/software/SfM/文件夹中, 在终端运行\$ python tutorial_demo.py
- 切换到tutorial_out/reconstruction_global目录: \$ pmvs2 ./PMVS/ pmvs_options.txt

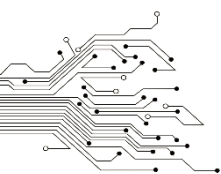


拓展--OpenMVG

---八斗人工智能，盗版必究---

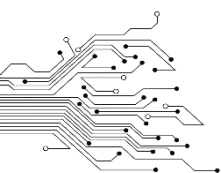
PMVS/models文件夹中生成一个大小为15.2MB的pmvs_options.txt.ply点云文件，用meshlab打开即可看到重建出来的彩色稠密点云：





拓展-SfM的劣势

- 1、尽管SfM在计算机视觉取得显著成果并应用，但是大多数SfM基于周围环境是静止这一假设，既相机是运动的，但是目标是静止的。当面对移动物体时，整体系统重建效果显著降低。
- 2、传统SfM基于目标为刚体的假设。



趋势1：多视觉几何与Deep learning方法融合

在深度学习一统天下的计算机视觉领域，三维视觉方向主导算法仍然是传统的多视角几何方法，但是深度学习方法也是一种重要而有效的辅助。

趋势2：多传感器融合

目前基于普通摄像头的视觉传感器仍然是主导，但是工业界对算法的鲁棒性要求比较高，纯视觉方法很难保证在复杂的环境下保持鲁棒的效果。所以，用廉价的激光传感器、IMU（惯性测量单元）等与视觉传感器进行融合，是一种比较靠谱的方法。比如在移动端目前基于摄像头+IMU方法越来越多。

趋势3：算法与硬件的结合

比如深度相机厂商在尽力把一些视觉算法嵌入到相机前端，提升相机的本地处理能力。

趋势4：算法与具体应用的结合能够更快的推动算法进步

三维视觉算法目前比较好的商业应用主要集中在无人驾驶、无人搬运车（Automatic Guided Vehicle，简称AGV）、AR（教育、影音游戏）、机器人等领域。



---八斗人工智能，盗版必究---

