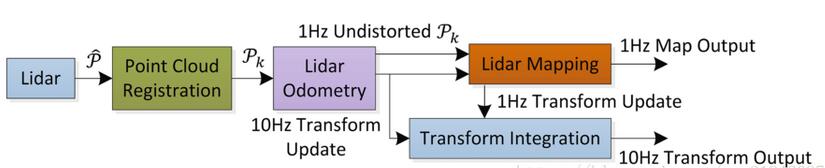
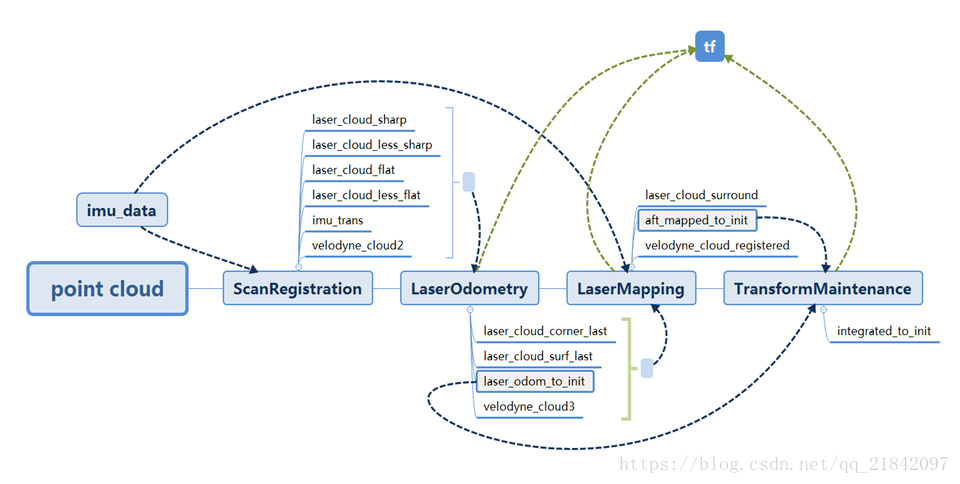
# 本周工作总结：





还是以这两个图为准

Lidar接收数据到后首先进行Point Cloud Registration线程，Lidar Odometry线程以10Hz的频率进行运动估计和坐标转换，Lidar Mapping线程以1Hz的频率构建三维地图，Transform Integration线程完成位姿的优化。这样并行的结构保证了系统的实时性。

整个算法分为四个模块，相对于其它直接匹配两个点云的算法，LOAM是通过提取特征点进行匹配之后计算坐标变换。具体流程为：ScanRegistration 提取特征点并排除瑕点；LaserOdometry从特征点中估计运动，然后整合数据发送给LaserMapping；LaserMapping输出的laser\_cloud\_surround为地图；TransformMaintenance订阅LaserOdometry与LaserMapping发布的Odometry消息，对位姿进行融合优化（先到的信息被利用，之后我们会详细再说）。

详细理解部分：

1文章中使用曲率公式来区分两种特征点。

文章中对曲率公式的解释为: smooth of the local surface 局部平面的光滑度。曲率公式本质可以表示为该点与其周围点距离的偏差：代码中取前后5个点(共10点)的偏差。可以想到：调整该处，可在运算速度和算法准确度之间做一个权衡。

2 对两类瑕点进行剔除。

针对论文的(a)情况，当某点及周围点据另外的点间的距离平方大于某阈值a（说明这两点有一定距离）且某点到其前后两点的距离均大于c倍的该点深度，则该点判定为不可标记特征点的点（入射角越小，点间距越大，即激光发射方向与投射到的平面越近似水平）。

针对论文的(b)情况，两向量夹角小于某阈值b时（夹角小就可能存在遮挡），将其一侧的临近6个点设为不可标记为特征点的点构建了一个等腰三角形的底向量，根据等腰三角形性质，判断X[i]向量与X[i+1]的夹角小于5.732度(threshold=0.1)，如depth1>depth2则X[i+1]距离更近，X[i]不标记为特征点；depth1<depth2 X[i]距离更近，X[i+1]不标记为特征点。

1. imu信息的获取

减去重力对imu的影响

解析出当前时刻的imu时间戳，角度以及各个轴的加速度

将加速度转换到世界坐标系轴下

进行航距推算，假定为匀速运动推算出当前时刻的位置)

推算当前时刻的速度信息

1. 利用提取到的特征点，建立相邻时间点云数据之间的关联，由此推断lidar的运动。

在这里，由于需要高速的处理，并没有采用一一对应的点云配准（事实上，对于点云来说，这也是很难做到的），而是在上一时刻寻找两点确定一条与当前时刻的角点距离最近的直线作为该角点的配准对应，同理，在上一时刻寻找三点确定一个与当前时刻的平面点距离最近的平面作为该平面点的配准对应。本质上匹配公式也是距离的计算公式，于是整个配准过程变成了优化过程——不断迭代使得距离之和最小。

之后就是一系列的优化操作：包括矩阵内插公式，刚体的旋转平移变化公式，矩阵求逆等，进而估计机器人位姿和运动。

# 下周工作计划：

# 继续注释LOAM算法剩余部分代码。