1.Init后需要传输的数据:

points\_old points\_centerold

mIQua mCQua

maskC--标记在框内的点 maskBC--标记暴匹的点

img\_old gray\_old

status=1

2.BF时重新初始化的步骤:

和第一次的ORB点匹配 得到暴匹点对应的points\_centernew

计算points\_centernew的质心,按第一次的大小和比例取大框和小框 得到mIQua mCQua

重新在大框内提点得到points\_old

points\_old在小框中的点设为points\_centerold 且标记maskC

当前帧图像设置为img\_old gray\_old

status=1 返回OF

3.光流如流程图PDF

结束后status=1

进BF边界条件:小框内点太少<=1;运动过快光流丢点过多,存活率<0.2 --status=2

6.22 讨论：

1. 给锚点判断边界，出去后close

2. 决定初始帧和后续帧的size：

都是给480X360的大图，输出的锚点是否在边界内由前段判断，这样就解决了锚点在大图内而不在边界内的情况，还是可以继续追

思考：服务器命中后，给的框是相对于初始帧的。如果命中后的当前帧和初始帧之间运动过大，要马上全图暴匹，搜索出新的框，这时候会卡吗？

都用大图的话，计算量会比小图大

需要测试修改的数据：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据 | Init时ORB点数 | Init时判断特征点过少的阈值 | 光流迭代次数和参数 | 光流存活率 | 大框太小需要补点的条件 | 扩框 | 最多增点数 | 删除距离过大的点 | 不稳定时暴匹尝试次数 |
| now | 500 | 5 | / | 0.2 | 长或宽缩小一半或点数少于ThP=250，且点数小于500 | 大框扩2，小框扩1 | 大框MAX\_COUNT=100，小框CMAX\_COUNT=20; | 4倍平均距离 | 50次后失败 |