

1 代码见附件，结果如下

不固定第一帧和第二帧

```
zy@zy-ThinkPad-E490 ~/Downloads/hw_course5_new/build/app $ ./testMonoBA
0 order: 0
1 order: 6
2 order: 12

ordered_landmark_vertices_size : 20
iter: 0 , chi= 5.35099 , Lambda= 0.00597396
iter: 1 , chi= 0.0289048 , Lambda= 0.00199132
iter: 2 , chi= 0.000109162 , Lambda= 0.000663774
problem solve cost: 0.369397 ms
makeHessian cost: 0.218024 ms

Compare MonoBA results after opt...
after opt, point 0 : gt 0.220938 ,noise 0.227057 ,opt 0.220992
after opt, point 1 : gt 0.234336 ,noise 0.314411 ,opt 0.234854
after opt, point 2 : gt 0.142336 ,noise 0.129703 ,opt 0.142666
after opt, point 3 : gt 0.214315 ,noise 0.278486 ,opt 0.214502
after opt, point 4 : gt 0.130629 ,noise 0.130064 ,opt 0.130562
after opt, point 5 : gt 0.191377 ,noise 0.167501 ,opt 0.191892
after opt, point 6 : gt 0.166836 ,noise 0.165906 ,opt 0.167247
after opt, point 7 : gt 0.201627 ,noise 0.225581 ,opt 0.202172
after opt, point 8 : gt 0.167953 ,noise 0.155846 ,opt 0.168029
after opt, point 9 : gt 0.21891 ,noise 0.209697 ,opt 0.219314
after opt, point 10 : gt 0.205719 ,noise 0.14315 ,opt 0.205995
after opt, point 11 : gt 0.127916 ,noise 0.122109 ,opt 0.127908
after opt, point 12 : gt 0.167904 ,noise 0.143334 ,opt 0.168228
after opt, point 13 : gt 0.216712 ,noise 0.18526 ,opt 0.216866
after opt, point 14 : gt 0.180009 ,noise 0.184249 ,opt 0.180036
after opt, point 15 : gt 0.226935 ,noise 0.245716 ,opt 0.227491
after opt, point 16 : gt 0.157432 ,noise 0.176529 ,opt 0.157589
after opt, point 17 : gt 0.182452 ,noise 0.14729 ,opt 0.182444
after opt, point 18 : gt 0.155701 ,noise 0.182258 ,opt 0.155769
after opt, point 19 : gt 0.14646 ,noise 0.240649 ,opt 0.14677
----- pose translation -----
translation after opt: 0 :-0.00047801 0.00115904 0.000366507 || gt: 0 0 0
translation after opt: 1 :-1.06959 4.00018 0.863877 || gt: -1.0718 4 0.866025
translation after opt: 2 :-4.00232 6.92678 0.867244 || gt: -4 6.9282 0.866025
```

固定第一帧和第二帧

```

zy@zy-ThinkPad-E490 ~/Downloads/hw_course5_new/build/app $ ./testMonoBA
0 order: 0
1 order: 6
2 order: 12

ordered_landmark_vertices_size : 20
iter: 0 , chi= 5.35099 , Lambda= 0.00597396
iter: 1 , chi= 0.0282599 , Lambda= 0.00199132
iter: 2 , chi= 0.000117497 , Lambda= 0.000663774
problem solve cost: 0.348206 ms
makeHessian cost: 0.162996 ms

Compare MonoBA results after opt...
after opt, point 0 : gt 0.220938 ,noise 0.227057 ,opt 0.220909
after opt, point 1 : gt 0.234336 ,noise 0.314411 ,opt 0.234374
after opt, point 2 : gt 0.142336 ,noise 0.129703 ,opt 0.142353
after opt, point 3 : gt 0.214315 ,noise 0.278486 ,opt 0.214501
after opt, point 4 : gt 0.130629 ,noise 0.130064 ,opt 0.130511
after opt, point 5 : gt 0.191377 ,noise 0.167501 ,opt 0.191539
after opt, point 6 : gt 0.166836 ,noise 0.165906 ,opt 0.166965
after opt, point 7 : gt 0.201627 ,noise 0.225581 ,opt 0.201859
after opt, point 8 : gt 0.167953 ,noise 0.155846 ,opt 0.167965
after opt, point 9 : gt 0.21891 ,noise 0.209697 ,opt 0.218834
after opt, point 10 : gt 0.205719 ,noise 0.14315 ,opt 0.205683
after opt, point 11 : gt 0.127916 ,noise 0.122109 ,opt 0.127751
after opt, point 12 : gt 0.167904 ,noise 0.143334 ,opt 0.167924
after opt, point 13 : gt 0.216712 ,noise 0.18526 ,opt 0.216885
after opt, point 14 : gt 0.180009 ,noise 0.184249 ,opt 0.179961
after opt, point 15 : gt 0.226935 ,noise 0.245716 ,opt 0.227114
after opt, point 16 : gt 0.157432 ,noise 0.176529 ,opt 0.157529
after opt, point 17 : gt 0.182452 ,noise 0.14729 ,opt 0.1823
after opt, point 18 : gt 0.155701 ,noise 0.182258 ,opt 0.155627
after opt, point 19 : gt 0.14646 ,noise 0.240649 ,opt 0.146533
----- pose translation -----
translation after opt: 0 :0 0 0 || gt: 0 0 0
translation after opt: 1 : -1.0718 4 0.866025 || gt: -1.0718 4 0.866025
translation after opt: 2 :-3.99917 6.92852 0.859878 || gt: -4 6.9282 0.866025

```

对比固定第一帧和第二帧前后的优化结果，优化完成后，两种方式得到的空间点的逆深度与真值都很相近，但是，固定前，第一帧相机 pose 的位置不再是(0, 0, 0)；而在固定后，使两帧位姿对应的雅克比为 0，信息矩阵中对应位置的矩阵块也为 0，利用 LM 算法求解时，相当于固定了第一帧和第二帧位姿不被优化，也就不会发生改变。

2 结果如下

```

----- TEST Marg: before marg-----
    100      -100      0
   -100  136.111 -11.1111
      0 -11.1111  11.1111
----- TEST Marg: 将变量移动到右下角-----
    100      0     -100
      0  11.1111 -11.1111
   -100 -11.1111  136.111
----- TEST Marg: after marg-----
   26.5306 -8.16327
  -8.16327  10.2041

```

可以看出，在 marg 变量 V2 之前，变量 V1 与变量 V3 信息矩阵对应位置为 0，即 V1 与 V3 关于 V2 条件独立，将 V2 移到右下角并 marg 之后，变量 V1 与变量 V3 产生联系，信息矩阵对应位置不再是 0，即 V2 被 marg 后将信息传递给了 V1 与 V3。

