关于矩阵：

分为四个象限。Top-Left=2,T-R=1,B-L=4,B-R=3。

不分了↑

Json文件说明：

totalLen是整个矩阵的宽度

groupLen是每个纤芯组的宽度

coreThre是判定在纤芯内部的阈值（百分比）。要求是一旦大于这个值，必须在纤芯内部。（过量判断）

coreLen是纤芯的宽度，通常大于groupLen的五分之一。（足量判断）。

coreLenThre是判定在纤芯内部的阈值（百分比）。要求是一旦在纤芯内部，必须大于这个值。（足量判断）

seStep：好像没用到

maxCore：最大值

disturange:确定最大值时，剩余点的个数小于disturange时开始地毯式扫描。

coreThreDis:拟合圆时，点的值与阈值差值小于coreThreDis则判定这点为圆周上的点。

start:搜索时的起始点序号（相对于maxPoint）

csvName:数据文件的名称，仅供测试使用。

Config.json存储可能需要改变的常量

totalLen是整个矩形的边长

coreThre是计算纤芯的光强亮度的阈值（某点如果超过这个光强，那么就当作纤芯内）

Locate.roughScan中更改步长可以改粗扫精度。

Bearing中，0是左，1是右下，2是右上。对数组而言，0是上，1是左下，2是右下

接口类完全崩坏了，反正你们到时候也要重写的，我就不管了。Query和GiveQuery都建议重写，具体怎么写你们怎么方便怎么来。

关于A\*：

每个open和closed中一个点对应一个list，依次存储[g(n)+h(n), g(n), [x, y], [lastX, lastY]]