**上次汇报：**

1.坐标一致性的问题

编辑很重要的一个是，屏幕在缩放平移旋转，这些变换对应到坐标系的变换，注意标定的结果只是对于标准图片的投影变换。

第一个问题，屏幕确定了一个坐标范围，在空间范围中要去查询摄像头，寻找的摄像机都是在这个屏幕范围内的。

第二个问题，屏幕的范围和摄像机都定了，再平移旋转缩放，实际上就是确定的摄像机四边形和屏幕结点的关系。

2.节点的分类、定义，其不变量，可变量的描述方法

最后结果是要要形成图（图论中的图）。要把图建立起来，需要定义节点，包括摄像机（切分）结点、电视墙（重组）结点。

结点里面是有内容的。例如切分节点，一个是位置，一个是视场（能照到哪个范围，例如可以用四个点，但是这四个点是什么坐标呢）。屏幕结点实际上是由地图数据决定的，屏幕中开的窗口在哪个空间范围。实际上每个结点又是一些数据，结点数量是定的，描述结点的属性是不定的。

3.连结的边如何确定

定义好两类节点后还不是图，还要定义边的相互连接关系，从而确定相连的边，采用可计算的方法确定空间是否有交集（有交集，边就相连，这个交集说的应该是摄像机节点和电视墙节点的交集），对于两个四边形有没有交集这个问题我们要具体化。

总结起来，形成图要分两步，第一步，先确定结点，第二步，求交叠。

**本周工作：**

1. 坐标一致性的问题

数据库中提供的摄像头参数是对于标准图片的投影变换。

用户可以缩放平移旋转全景范围查询摄像机。

数据库返回摄像头参数。



也就是数据库提供的是一系列相对标准图片坐标系的坐标点。



把范围告诉数据库，并查询出相对坐标。



另外一种方式，数据库把所有数据发给分布式处理中心。

对于坐标的平移缩放旋转不知道是不是用仿射变换进行处理。

2.节点分为两类：切分节点、重组节点

## 切分节点

（接收图像，处理图像，切分，发送）

视频流及相关信息（摄像机提供）

投影变换矩阵

切分信息：需要切分的横纵坐标

拓扑信息：本节点切分后每一块发送到哪一个重组节点

## 重组节点

单个电视墙屏幕分辨率宽、高

拓扑信息：本重组节点接受哪几个切分节点发送的视频流

位置信息：切分节点发送的视频流小块在屏幕中的坐标位置

重叠区域：切分节点发送的视频流有无重叠区域，重叠区域坐标

重叠区域的融合系数

3. 一些计算机几何的开源库

**a)** **Geos**

Geos的全称就是Geometry Engine Open Source，参考网站就是：[http://geos.refractions.net](http://geos.refractions.net/)，从名称就可以看出，Geos就是判断几何体的过程。

**b)** **CGAL**

CGAL的全称就是Computational Geometry Algorithms Library，参考网站就是：[http://www.cgal.org](http://geos.refractions.net/)，从名称就可以看出，CGAL就是计算几何的开源库。

**c)** **LEDA**

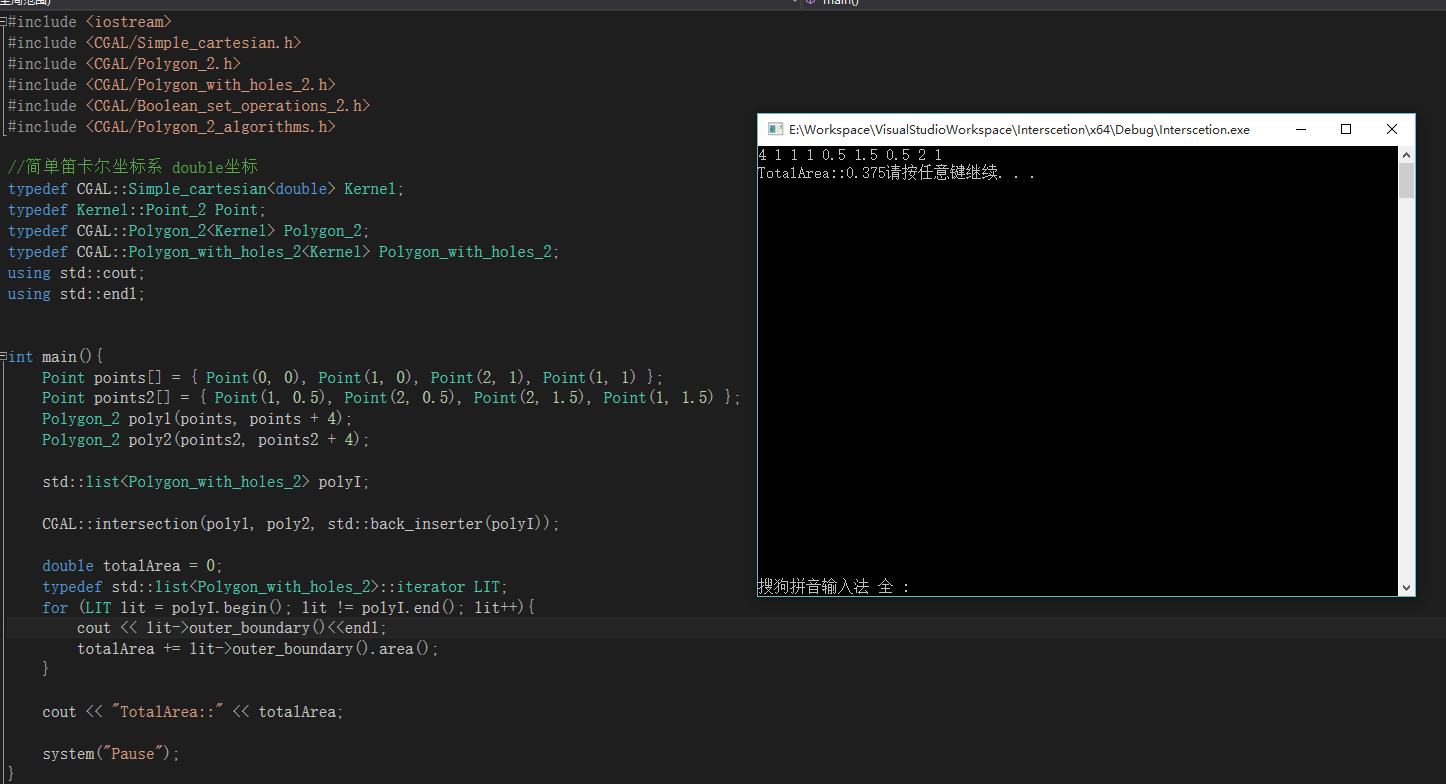
LEDA的全称就是Library of Efficient Data types and Algorithms，参考网站就是：<http://www.algorithmic-solutions.com/>，

**d)** **Wykobi**

Wykobi就是指的是Wykobi Computational Geometry Library ，参考网站就是：[http://www.wykobi.com](http://www.wykobi.com/) ，这个用C++语言实现的开源库，功能比较全，但是实现思想比较传统，也是比较简单。可以作为研究用，但是作为项目参考意义就不是很大。

选择了CGAL库进行测试。

配置CGAL主要使用[CMake](https://cmake.org/)、[CGAL installer](http://www.cgal.org/download/last) 、Qt 5、libQGLViewer、Boost。



涉及到的问题： 图形的几何变换

几何变换改变图像中像素的空间关系，或者说是像素的移动。

由两个组成部分

（1）坐标的空间变化；

（2）灰度内插，即对空间变换后的像素赋值

像素通过变换映射到新的坐标位置，新的位置可能是在几个像素之间，即不一定为整数坐标。这时就需要灰度级差值将映射的新坐标匹配到输出像素之间。

对于平面区域，有两种方式的几何变换：一种是基于2\*3矩阵进行的变换，叫仿射变换；另一种是基于3\*3矩阵的变换，又称透视变换或者单应性映射。可以把后一种变换当作一个三维平面被一个特定观察者感知的计算方法，而该观察者也许不是垂直观察该平面。

最常用的空间坐标变换之一是仿射变换，提供了一个把一系列操作连接在一起的框架。把一幅图像上的像素重新定位到一个新位置。

一个坐标通过函数变换的新的坐标位置：

C:\Users\Administrator\AppData\Local\YNote\data\etet2007@163.com\20af39cb767048f88f83fcc46133a71d\8357037_3033.png

所以在程序中我们可以使用一个2\*3的数组结构来存储变换矩阵：

C:\Users\Administrator\AppData\Local\YNote\data\etet2007@163.com\6ac41794486a4bd08f72cd9cdb39d0c5\8357084_2179.png

以最简单的平移变换为例，平移（b1,b2）坐标可以表示为：

C:\Users\Administrator\AppData\Local\YNote\data\etet2007@163.com\61499a56e1184d98b5ee5eb3d6c5f435\8357151_4157.png

因此，平移变换的变换矩阵及逆矩阵记为：

C:\Users\Administrator\AppData\Local\YNote\data\etet2007@163.com\b9b4d33e9a364a57b04dad292cdebb78\8357196_5767.png

缩放变换：将图像横坐标放大（或缩小）Sx倍，纵坐标放大（或缩小）Sy倍，变换矩阵及逆矩阵为：

C:\Users\Administrator\AppData\Local\YNote\data\etet2007@163.com\6be40f9b55454272ac7053da7ce9d792\8357283_1562.png

选择变换：图像绕原点逆时针旋转a角，其变换矩阵及逆矩阵（顺时针选择）为：

C:\Users\Administrator\AppData\Local\YNote\data\etet2007@163.com\3ff7ccc310f64668aad0fcf4b7bb63ad\clipboard.png