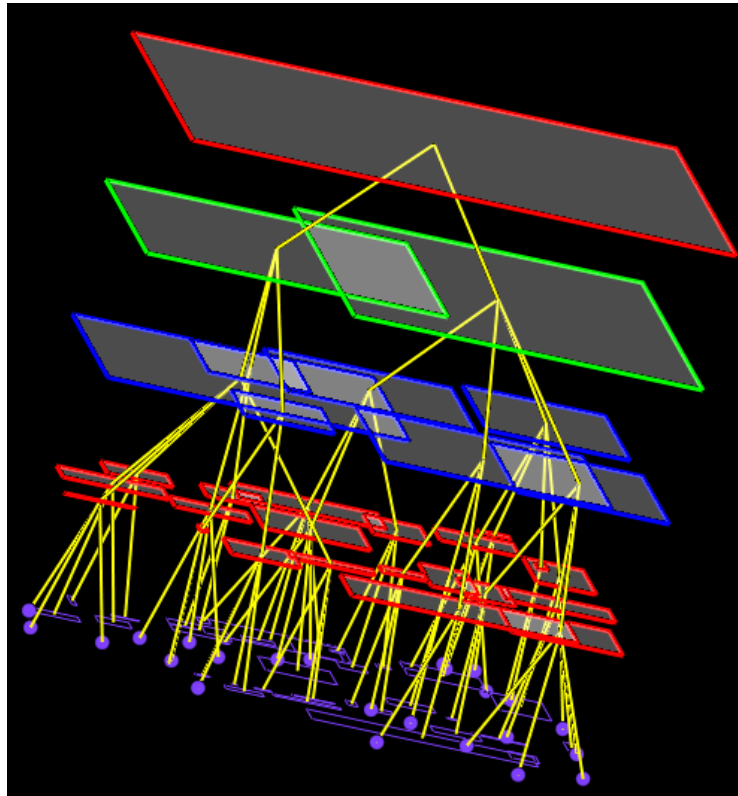


R-树及其演示程序使用说明



一、 程序说明

本实验完成的是 R-Tree 和 R*Tree 的实现，这两种数据结构可以实现任意多维数据的建树、查找、插入和删除操作。如果想使用 R-Tree API，请按该文档的第二部分《如何使用 R-Tree》将对应的文件添加到您的程序框架中。

本实验所提供的可执行程序是对二维数据 R-Tree 和 R*Tree 的演示，更高维度的数据无法给出直观显示。可执行程序只是辅助用户对 R-Tree 有更直观的了解，演示程序中所演示的功能也仅仅是 R-Tree 的一部分。

二、 如何使用 R-Tree

关于 R-Tree 的定义以及介绍，请参考实验报告部分或参考文献，这里不再赘述。本次实验的工程中包含两部分，一部分是 R 树的实现，另一部分是对二维 R 树的直观演示程序，采用 MFC 实现。如果想在您的程序中使用 R 树进行查找结构的建立和使用，请抽取以下文件到您的工程中：

RSTDefine.h 该文件定义了一些基本函数以及 R 树中的数据接口，使用 R 树进行查找结构的建立和使用必须实现我们提供的数据结构，使您的数据继承自类 RSTNode，并实现其中的一些操作，例如返回数据在各个维度上的区间范围等。RSTTestData.h 中已经根据 RSTDefine.h 中的接口实现了二维点数据和二维矩形数据，可以参考其中的实现对您的数据进行封装。


RSTBasisFunction.cpp 该文件中包含了 RSTDefine.h 中的一些函数定义。

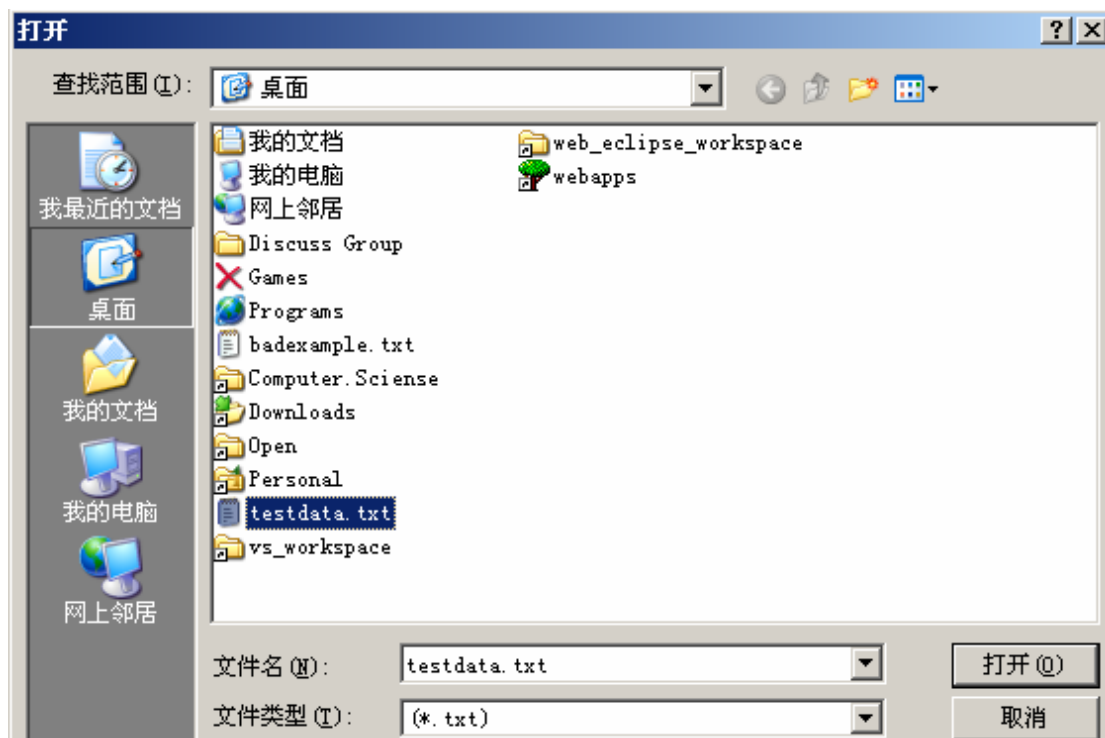
RSTRTree.h 和 **RSTRTree.cpp** 实现了 R-树和 R*树，其内部提供了插入数据、删除数据、查找数据等操作。详细情况请参考代码注释和参考文献。

三、 如何使用 R-Tree 演示程序

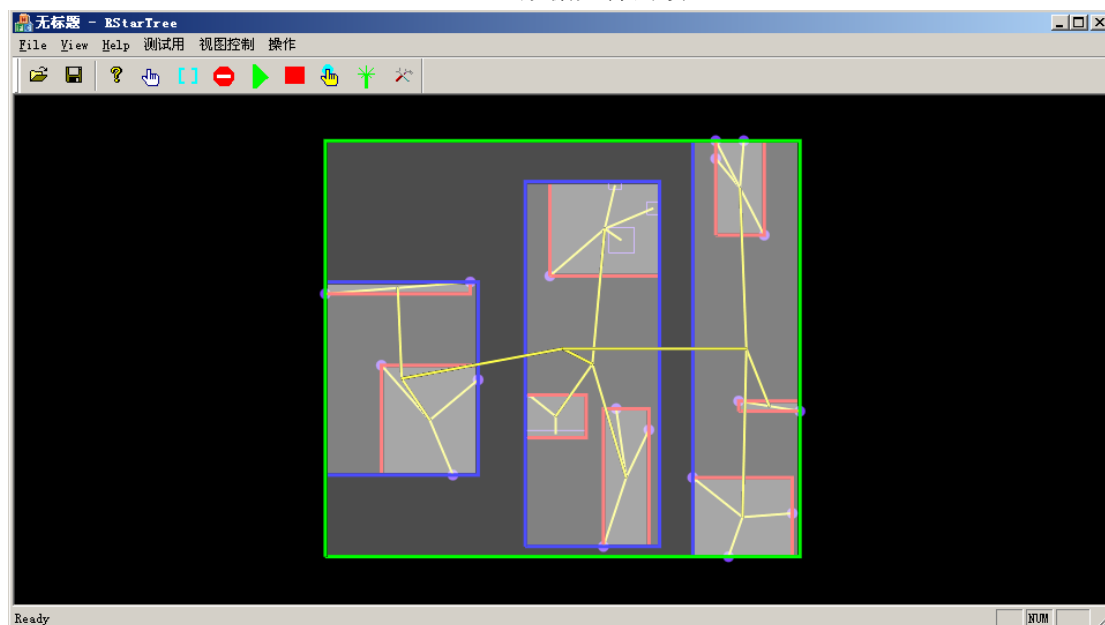
演示程序提供了二维数据构成的 R 树的直观显示和操作，由于 OpenGL 最多提供三维数据演示，因此，演示程序只接收二维数据输入。**运行程序，将鼠标停留在某个工具栏按钮上，会显示相应的按钮功能说明。**

1. 数据输入



数据的输入可通过两种途径，**第一种是文件输入**，文件的格式请参考样例文件或实验报告中关于数据格式的说明，输入方式为文件->打开，或者点击界面上的  按钮。打开文件后程序会读取文件内容，并建立好 R 树并演示出来。输入完毕后，**按住鼠标左键拖动实现平移，按住鼠标右键拖动实现旋转。**

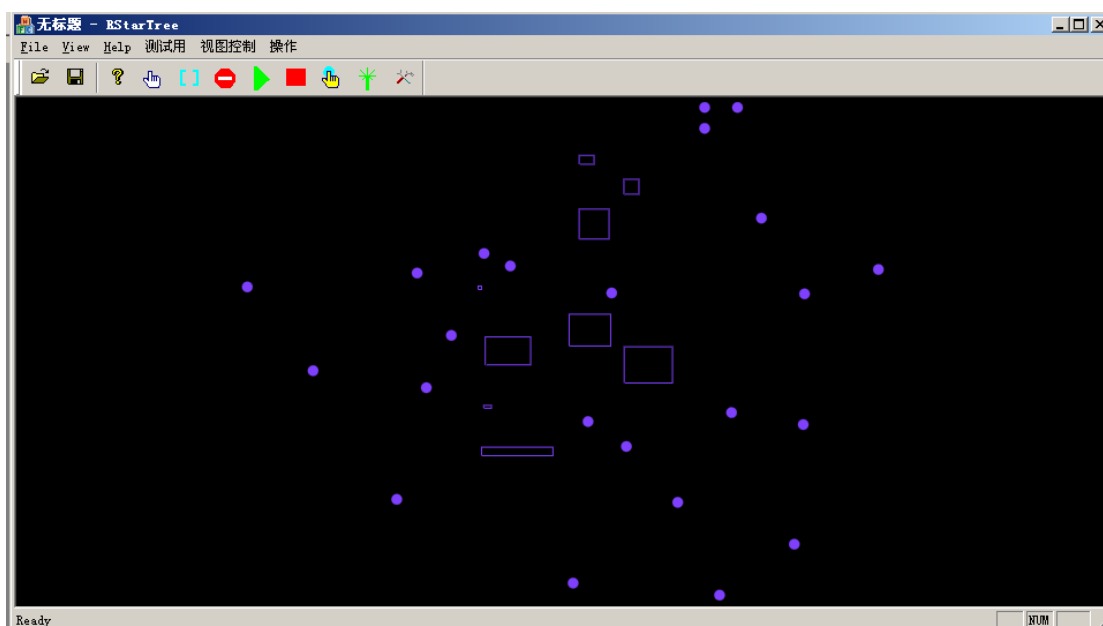


数据文件的读入





读入数据后可进行树的演示

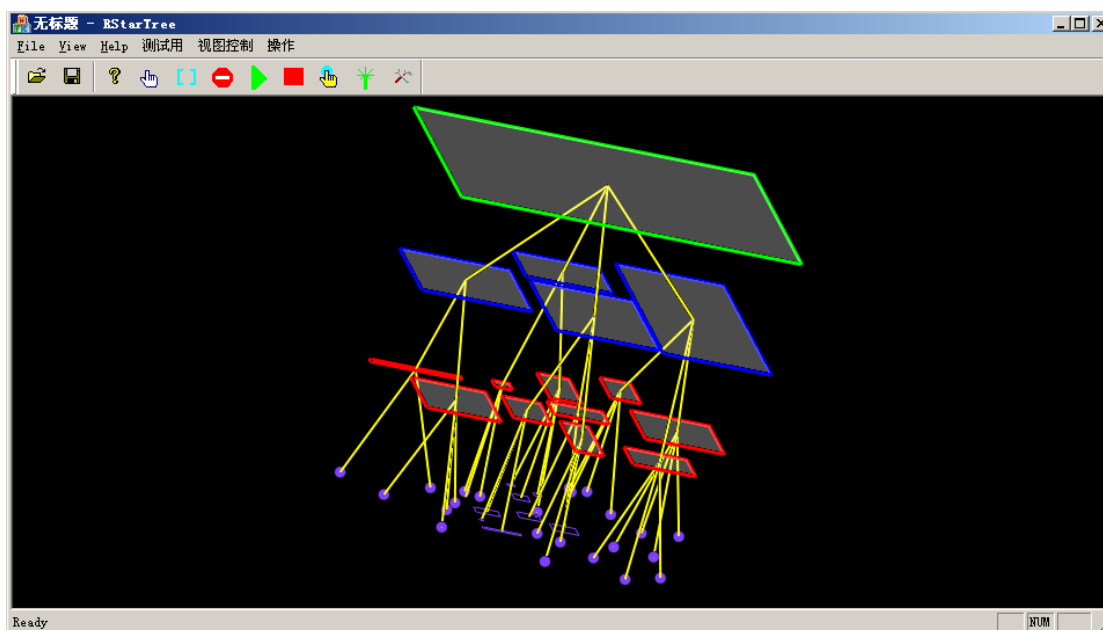
第二种输入方式为手动加入二维点数据和矩形数据，点击  按钮，然后可以单击鼠标左键加入点数据，或按住鼠标左键拖拽添加矩形数据，如下图所示。手动输入数据完毕后，点击  按钮可以显示建立好的 R 树。手动添加数据后，屏幕可能会出现抖动，这是因为程序重新计算了现有数据的包围盒，并按比例重新显示，不影响数据的有效性，此时可以用鼠标滚轮调节显示比例，按住鼠标右键拖动。



手动输入数据

2. 树的显示以及鼠标控制

点击  显示树之后用户所看到的视图方向是从树根到树叶的正面视图，用户可以点击  实现用鼠标对显示的控制，**按住鼠标左键可以对树进行拖动，按住鼠标右键拖动可以旋转，调整树的角度，滚动鼠标滑轮可以进行放大和缩小。**如下图所示。





用鼠标控制树的显示

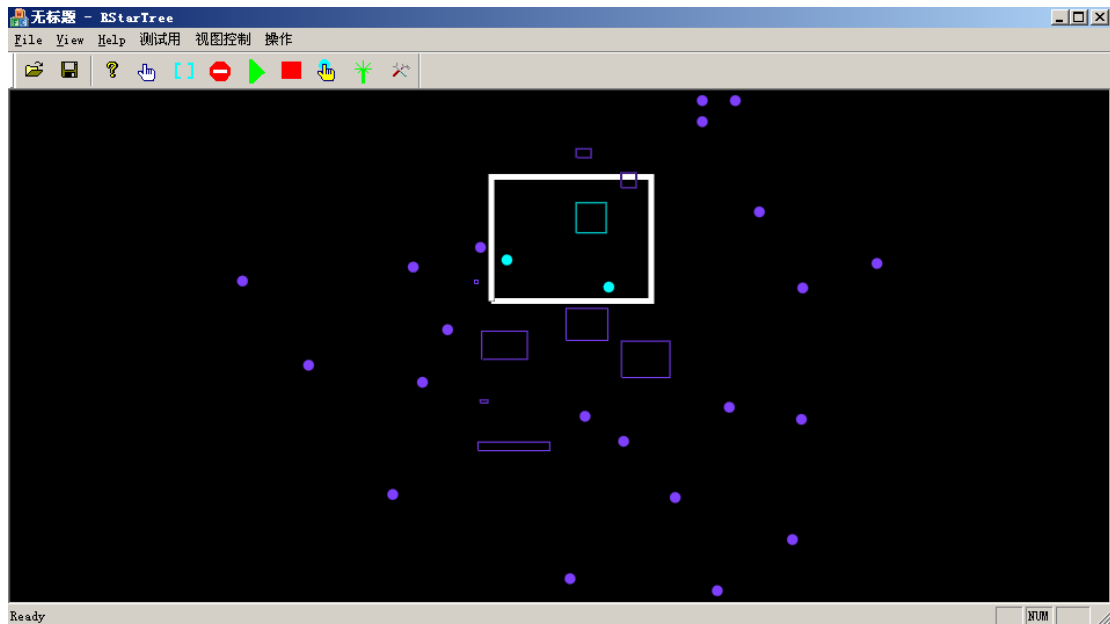
如上图所示，该 R 树分为 4 层，最底层为用户输入的二维点数据，最上层为树根，树根所显示的矩形刚好为输入数据的矩形包围盒。树的每个节点与其孩子的关系通过黄色的树枝显示。

同时，在进行数据的 **鼠标输入、查询和删除时，鼠标左键通常作为功能键，但此时仍**

然可以通过滚动鼠标滑轮对视图进行放大和缩小，按住鼠标右键可以对当前数据进行拖动。


3. 二维区间查找

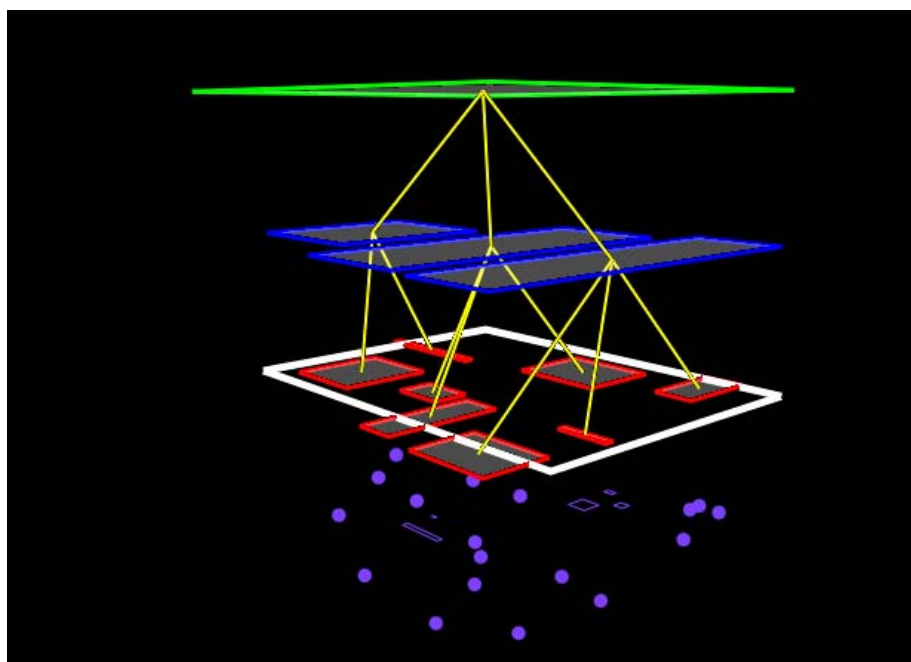
点击  按钮可对树中的数据进行二维查找。点击  后，视图会自动切换，只显示用户输入的数据，如图所示。









按住鼠标左键，拖动鼠标可以确定二维查找矩形，程序会通过 R 树的查找算法查找完全包含在查找矩形内的数据，并将其显示。注意，在拖动鼠标构造查询矩形的时候，请不要将鼠标拖出客户区外，否则这个查询矩形是无效的。

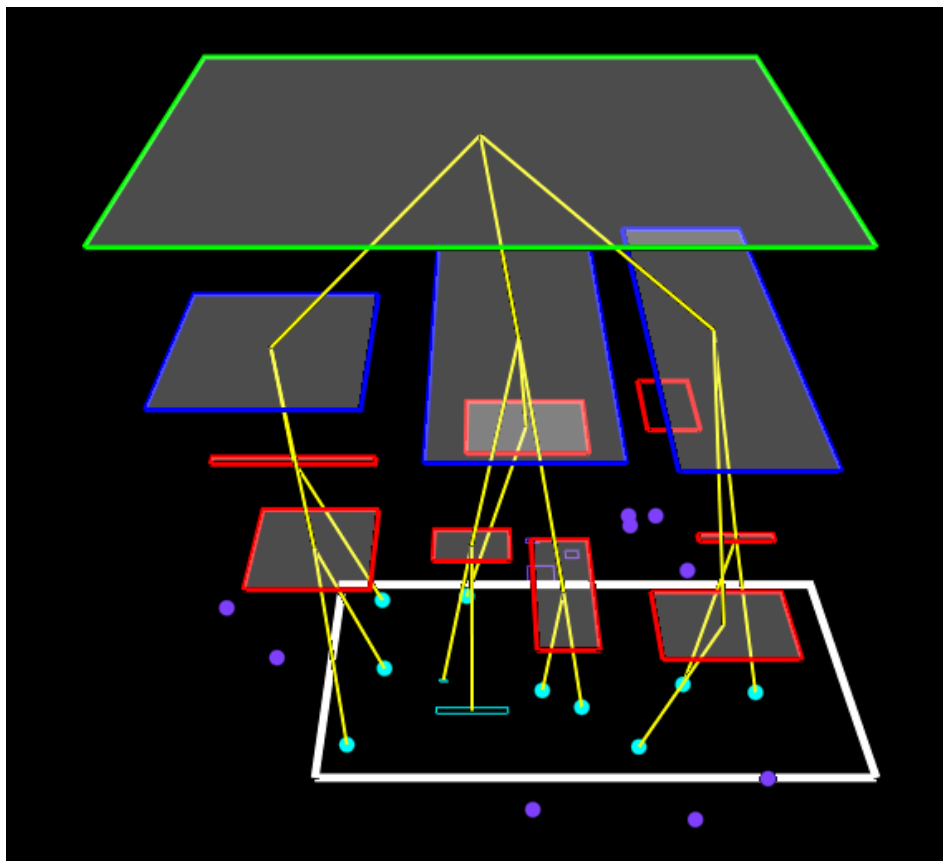
4. 查找过程演示

在二维查找的基础之上，可以点击  按钮，进行查找过程的演示，演示过程将给出在 R 树上每一层的查找结果，查找过程中被命中的矩形会带有半透明膜，以表示与其他矩形的不同，同时，查找路径中所涉及到的树枝也会相应的显示出来。如下图所示。



查找过程演示

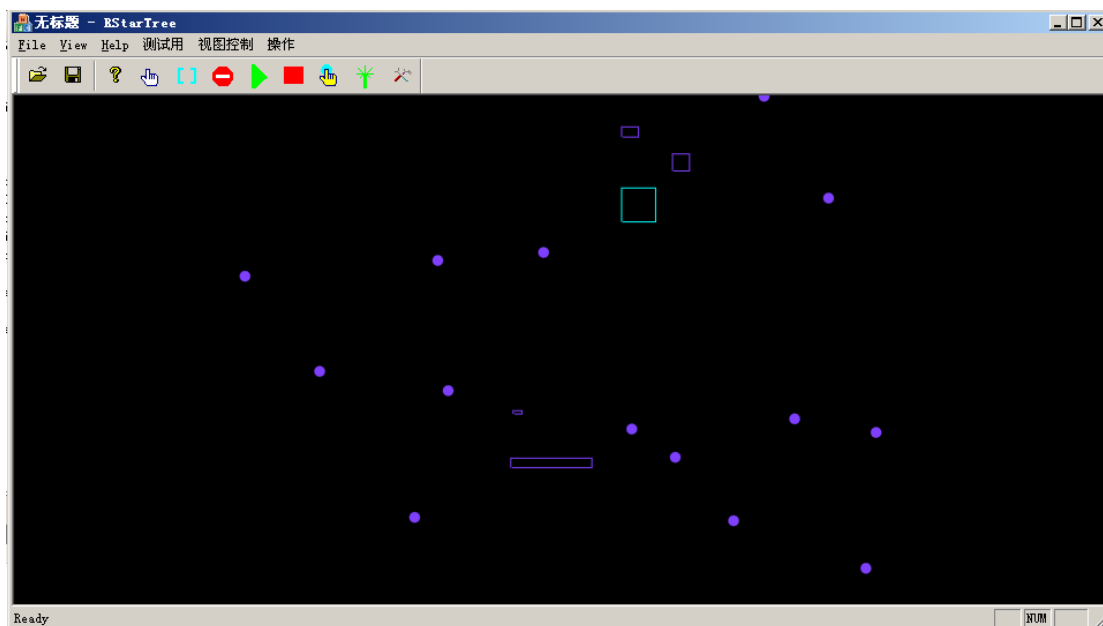
其中，白色的矩形为用户的查找矩形。在演示过程中的任何时候，可以点击  按钮停止演示过程，也可以点击 、 和  等按钮中断演示过程，进行相应的操作。演示结束或被终止后，可以通过点击  对显示对象进行控制，也可以点击  重新显示 R 树。演示结束后，白色查找矩形会落在数据层，完全包含在查找矩形内的数据会变色显示，如下图所示。



查找完毕



5. 区域定位

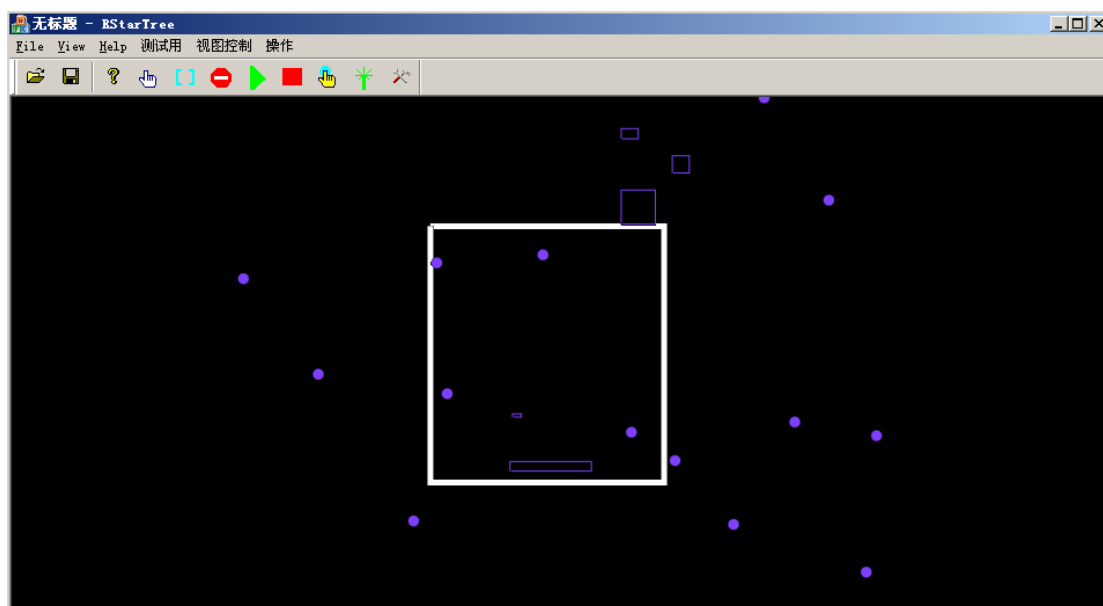
区域定位是确定用户指定的一点位于平面内的何处，R 树支持区域定位操作，对于用户输入的点，程序会报告出该点位于哪个矩形内。在演示程序菜单中点击“操作->区域定位”，视图会自动切换到只显示数据的状态，用鼠标左键点击屏幕，程序会报告点所在的矩形，如果不在任何矩形内，则不进行提示，如下图所示。



区域定位演示


6. 数据删除

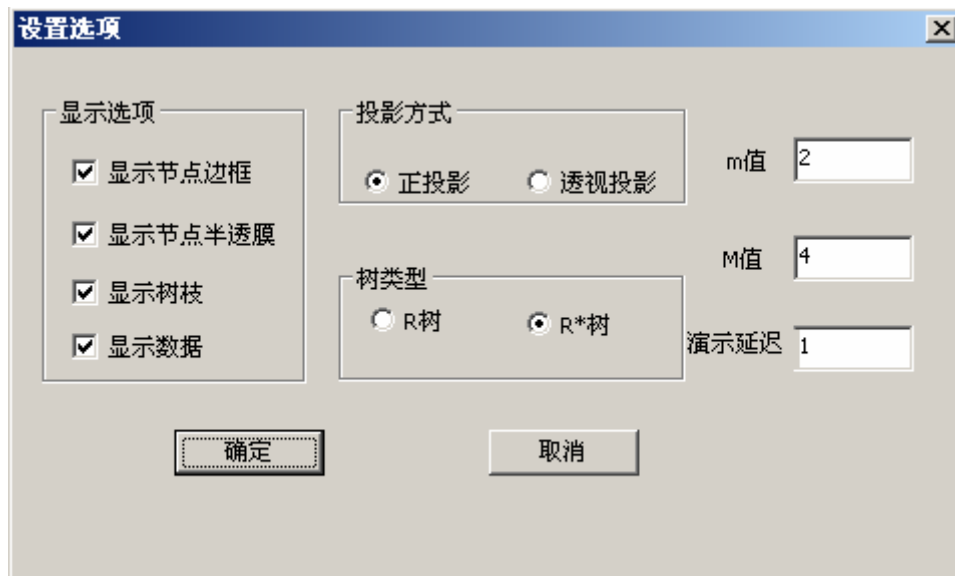
对于现有的数据，可以点击  按钮进行数据的删除，点击  后，会切换视图，只显示数据，此时用户可以拖动鼠标左键构造矩形，完全包含在矩形内部的数据会被删除。数据删除后屏幕会出现一定的抖动，这是因为程序重新计算了删除后剩余数据的包围盒，不影响数据的有效性。删除过程如下图所示。



删除过程演示

7. 控制面板设置

点击  按钮可调出控制面板，进行一些参数的设置。如下图所示。



控制面板

显示选项设置显示树的时候的可选项。投影方式控制 OpenGL 显示的投影方式。树类型表示演示程序内部所采用的 R 树类型，类型更改后会在下次读入数据建立 R 树时生效。m 值和 M 值分别表示树节点中孩子个数的下限和上限。演示延迟控制查找演示动画速度，默认为 1 毫秒，数值越大，演示过程进行的越慢。