

文章编号: 1005-8451 (2007) 10-0004-03

应用 ObjectARX 开发地形图的窗口剪切程序

程亚飞, 王卫东

(中南大学 土木建筑学院, 长沙 410075)

摘要: 为了对公路或铁路线路大幅带状地形图中的直线、多义线、样条曲线、圆、填充和块等各种实体方便地进行剪切操作, 利用 ObjectARX 在 VC++ 平台上开发了窗口剪切程序。利用此工具可以保留任意形状窗口内的实体, 删除窗口外的其余部分, 且不改变实体的原有属性。不论是开环还是闭环的线状实体, 均利用 getSplitCurves() 对线状实体重画, 同时完善了对子曲线进行取舍操作的算法。

关键词: ObjectARX; 实体剪切; 地形图; 窗口剪切程序

中图分类号: TP391.72

文献标识码: A

Application of ObjectARX to developing Window clipping program for topographic maps

CHENG Ya-fei, WANG Wei-dong

(Civil Architectural Engineering College, Central South University, Changsha 410075, China)

Abstract: To cut entities of long ribbon of topographic maps of road or railway, such as straight lines (AcDbLine), polylines, spline curves, circles, paddings (AcDbHatch), block (AcDbBlock) and so on, it was developed the window clipping program based on ObjectARX in the developing environment of Visual C++6.0. Using this tool, we could reserve the entities keeping its attribute within the arbitrary shape window and delete the parts which were out of the window after having broken the curve at the point of intersection. To striation entities, closed or opened, the function of getSplitCurves() could re-draw the reserved entities. Also the method of reserving or deleting the sub-curves was been discussed.

Key words: ObjectARX; entity clipping; topographic map; window clipping program

在公路和铁路的线路设计中, 需要经常对大幅带状的地形图进行任意形状边界的剪裁操作, 将边界内实体提取出来成为另一幅图。运用专业的大型软件进行此项操作比较复杂, 本文利用成熟、高效的 AutoCAD 二次开发工具 ObjectARX 开发了一个窗口剪切程序。该工具在 AutoCAD 中注册为外部函数命令, 可直接在 AutoCAD 中调用, 提高了工作效率。

1 基本思路

地形图中的实体主要是多义线、填充、直线、块、文字和圆等, 本文以这几种实体为剪切对象。

首先遍历地形图中所有的实体, 并得到其 ID。然后根据提示, 在屏幕上输入或选择多个点, 生成一个闭合的 AcDbPolyline 类对象作为剪切边界。最后依据不同种类的实体对象进行不同方法的剪切操作。

1.1 对曲线的剪切操作的基本思路及步骤

(1) 求出剪切边界与被剪切曲线的交点并按各交点与曲线起点的距离远近对交点排序; (2) 将原曲线重画 (在排好序的各交点处断开) 形成一系列首尾相连的子曲线; (3) 判断各子曲线的中点是否在闭合剪切边界内, 若在边界内, 将此段子曲线添加到当前数据库中; (4) 最后将原曲线从数据库中删除。

1.2 构造一个新的边界的填充实体

(1) 将剪切边界与原填充实体的最外层边界生成面域 AcDbRegion 类对象; (2) 利用面域的布尔运算求两者的面域交集作为新填充实体对象的外层边界; (3) 依次将原填充实体的各内边界所生成的面域与剪切边界生成的面域的交集添加到新填充实体中作为内层边界; (4) 按照原填充实体的属性对新填充实体进行填充; (5) 最后删除原填充实体。下面分别介绍对各种实体的具体剪切过程。

2 剪切曲线实体

2.1 普通曲线实体与剪切边界曲线的交点求法

收稿日期: 2007-02-03

作者简介: 程亚飞, 在读硕士研究生; 王卫东, 副教授。

除了对 `AcDb3dPolyline` 类对象外常见的曲线实体, 包括直线、圆、样条曲线和椭圆等, 都可以将剪切边界所在平面生成一 `AcGePlane` 平面后直接调用函数 `AcDbEntity::intersectWith()` 求交点, 即将被剪实体投影到剪切边界所在的平面上后求剪切边界与投影实体的交点, 再将交点返回原被剪曲线上, 最后得到的交点即是被剪实体上的点, 这些操作的实现都由函数 `intersectWith()` 自动完成。

2.2 `AcDb3dPolyline` 类曲线与剪切边界曲线的交点求法

对 `AcDb3dPolyline` 类曲线, 需先将被剪曲线投影到剪切边界所在平面, 再调用 `intersectWith()` 函数求出剪切边界与投影体的交点。利用投影体上某点的 `Param` 值 (由 `getParamAtPoint()` 得到) 与原被剪曲线上对应点的 `Param` 值相等的原理, 可以将求得的交点返回到原被剪曲线上。简略代码如下:

```
AcGePlane plane; // 定义剪切边界所在平面
AcDbCurve * projCurve; // 定义被剪曲线在剪切边界所在平面上的投影曲线
p3dPolyline -> getProjectedCurve (plane, normal, projCurve); // 得到投影曲线, normal 为投影平面的法向量
```

```
projCurve -> intersectWith (pPolyline, AcDb::kOnBothOperands, ptArr, 0, 0); // ptArr 为求得的投影曲线与剪切边界曲线的交点数组, pPolyline 为剪切边界曲线
```

// 将交点返回到原 3 维多义线上, `temptArr[i]` 即为最终得到的原 3 维多义线上的点

```
projCurve -> getParamAtPoint (ptArr[i], douArr[i]); // 得到投影体上对应某交点的 Param 值
```

// 根据 `Param` 值得到原被剪曲线上的点, 即为剪切边界曲线与被剪曲线的交点

```
p3dPolyline -> getPointAtParam (douArr[i], temptArr[i]);
```

2.3 交点排序及曲线重画

求得被剪曲线与剪切边界的交点后, 即可用 `AcDbCurve::getDistAtPoint()` 函数得到各交点与被剪曲线起点 (由函数 `AcDbCurve::getStartPoint()` 获取, 圆和椭圆也有起点) 的距离, 并按照此距离远近对交点进行排序。

在剪切程序中, `AcDbCurve::getSplitCurves()` 函数是一个很重要的函数, 不管是开环还是闭环的曲线, 均可用该函数根据排好序的点数组或参数数组

重画原曲线。若对闭环实体采用生成面域进行布尔运算的方法, 则需要在 `AcDb` 类库和 `AcGe` 类库之间相互转化, 过程十分复杂, 而且用面域操作时程序运行速度慢, 而用 `getSplitCurves()` 重画则比较直观, 而且被剪实体的属性不会有任何改变。

2.4 子曲线取舍算法的完善

关于子曲线的取舍算法需加以完善。以前多数文献中提到的方法是首先判断出曲线端点在剪切区域内部或外部, 再按子曲线序号的奇偶性决定其取舍。由于这个方法不能处理剪切边界曲线的顶点刚好落在被剪实体上这一特殊情况, 所以这个算法并不完善, 本文认为对每一段子曲线都应判断其与剪切边界的相对位置。

合理的算法是得到每一段子曲线的中点, 再判断此点与剪切区域的相对位置, 若此点在边界内, 则相应的子曲线就在边界内。判断时利用射线法, 即以需判断的点为基点, 构造一射线, 根据射线与剪切边界曲线的交点个数判断二者相对位置 (奇数在内, 偶数在外)。若射线刚好经过剪切边界曲线的顶点, 则应重新构造射线进行判断, 直至所构造的射线不经过剪切边界曲线的任何一个顶点。简略代码如下:

```
int ptOutOrIn(AcGePoint3d Point)
{ int num; // 剪切边界曲线与所构造的射线的交点个数
  for(int i=0; i++)
  { // 构造射线, 基点为需要判断的点, 射线的向量分量设为 i 的线性函数 (省略)
    AcDbRay * pDbRay = new AcDbRay;
    Int bol=0;
    for(int j=0; j<n; j++) // n 为剪切边界多义线 pPolyline 的顶点个数
    { // 将上一步骤中构造的射线转化为 AcGe 几何类用以判断剪切边界曲线的顶点是否在此射线上
      AcGeRay2d GeRay;
      if(GeRay.isOn(PArr[j], tol) != Adesk::kTrue) // 如果顶点 PArr[j] 不在构造的射线上。其中 tol 为 AcGeTol 类对象
        bol=bol+1;
    }
    if(bol==n) // 如果射线不通过剪切边界多义线的任何一个顶点则退出循环体
    { pDbRay -> intersectWith(pPolyline, AcDb::
```

```
kOnBothOperands,ptArr,0,0);
    num=ptArr.length();
    break;}
}
if((fmod(double(num),2.0)!=0)) return 0;
else return 1;
}
```

3 剪切填充实体

在线路平面图中，有较多的填充实体，如房屋和厂房等建筑物，所以有必要对填充实体做剪切操作。对有多个边界的填充实体的剪切操作的基本步骤是：(1) 首先将封闭的剪切边界曲线和被剪切的填充的外层边界实体分别生成面域，然后利用面域的布尔运算求此两面域的交集，此面域交集作为新填充实体的外层边界，重载函数 appendLoop() 进行添加；(2) 从外至内依次将各层内边界生成面域，分别求其与剪切边界生成的面域的交集，并依次用函数 insertLoopAt() 将各面域交集添加为至新的填充实体，做为新填充的各内层边界；(3) 按照原填充对象的填充属性（图案、样式、比例）对新填充实体进行填充；(4) 最后删除原填充实体。举例见图 1。

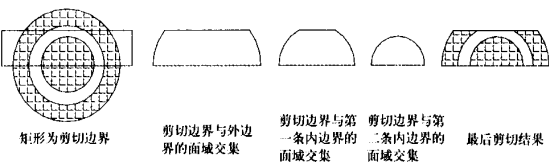


图1 填充实体的剪切变法

4 对文本的处理

对文本进行剪切没有任何意义，所以只根据其位置决定删除还是保留。对单行文本，不能准确求得文本范围 4 个角点的坐标，只能采用 AcDbExtents 类得到包围文本的长方形的最小点（长方形左下角的点）和最大点（长方形右上角的点），然后推得长方形的另外 2 个角点，当且仅当 4 个角点全部落在闭合剪切区域内时才将此文本保留，否则予以删除。对多行文本（如图 2），利用 getBoundingPoints() 得到文本覆盖范围的 4 个角点（如图中的 pt0~pt3），再

根据其实际宽度、旋转角度和对齐方式（用 attachment() 获得）将范围缩小，得到新的角点（如图中的 npt0~npt3），当且仅当 4 个新角点全部落在剪切区域内时才将此文本对象保留。

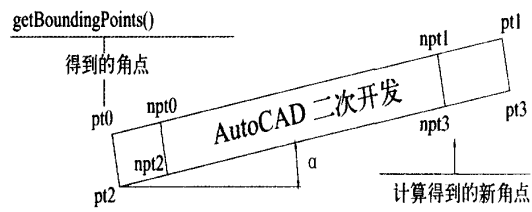


图2 多行文本剪切办法

5 对块的处理

由于块对象作为一个包含多个实体对象的块表记录，有其特殊性，在不解块的情况下对某一个块中的各个实体对象进行剪切后，与其相同的所有的块的形状都会随之而改变。所以仅将与剪切边界曲线相交的块解块后进行剪切，而对与剪切边界曲线不相交的块对象，若在边界内部则保留，否则就删除。

6 结束语

(1) 程序从实际应用出发，可以将标准图框放置在地形图上后选其 4 个顶点作为剪切边界多义线的顶点进行剪切，直接得到所需范围内的图形，提高了工作效率。(2) 窗口剪切的 ObjectARX 程序的各种算法已有很多，文章在已有算法基础上，提出自己新的算法，并对个别算法加以完善，缩短程序运算时间。

参考文献：

[1] 李世国. ActoCAD 高级开发技术 -ARX 编程及应用[M]. 北京：机械工业出版社，1999.
[2] 张忠泽. 在 AutoCAD 中实现图形的任意切边[J]. CAD/CAM 与制造业信息化. 2002 (6)：55-56.
[3] 梅树立，江景涛，陈忠良. 基于 AutoCAD 的任意复杂窗口剪裁技术[J]. 中国农业大学学报. 2001，6 (6)：77-80.
[4] 杜 刚，刘东学，张 磊. 基于 ObjectARX 的 AutoCAD 二次开发及应用实例[J]. 机械设计与制造. 2004 (3)：30-32.
[5] 邵俊昌，李旭东. AutoCAD ObjectARX2000 开发技术指南[M]. 北京：电子工业出版社，2000.