

文章编号: 0494-0911(2011)05-0013-03

中图分类号: P237

文献标识码: B

AutoCAD 中栅格图像定位研究

龚有亮¹, 翟 翊¹, 曹歆宏², 时 荣²

(1. 信息工程大学 测绘学院, 河南 郑州 450052; 2. 河南省遥感测绘院, 河南 郑州 450008)

Research on Raster Image Orientation in AutoCAD

GONG Youliang, ZHAI Yi, CAO Xinhong, SHI Rong

摘要:介绍栅格图像定位文件中各参数的含义,分析 AutoCAD 栅格图像功能的实现机制,着重讨论编程插入并正确定位栅格图像的原理和过程,并给出关键代码。

关键词: AutoCAD; 图像定位文件; 图像插入; 数字地图; 地图矢量化

一、引言

AutoCAD 是美国 Autodesk 公司推出的一种通用的、功能强大的计算机辅助设计软件,被广泛应用于建筑、机械、电子等许多领域。在测绘行业, AutoCAD 经过二次开发,成为地理信息数据采集和加工的平台,得到大规模的应用。在测绘生产实践中,经常需要把遥感图像插入到 dwg 格式的矢量地图中,使之成为地图数据的一部分,或者把纸质地图、调绘片等图件扫描纠正后形成栅格图像插入 dwg 文件中,作为地图矢量化的底图。但是, AutoCAD 本身的图像插入功能(imageattach 命令)比较弱,需要用户手工输入图像左下角点坐标、缩放比例和旋转角参数才能实现图像的正确定位,并且这些参数往往是未知的且不易求得。用各种 GIS 软件或遥感图像处理软件纠正后的栅格图像一般都带有一个定位文件,本文介绍利用 ObjectARX.NET 开发工具编程实现根据定位文件自动插入和定位栅格图像的方法。

二、栅格图像定位文件

存储栅格图像定位信息的方法^[1]一般有两种:

① 对于 tif、jpg 等通用图像格式,附加一个定位文件,定位文件的基本名与图像文件相同,但扩展名由图像文件扩展名的第一个和最后一个字母再加上字母 w 组成,比如 tif 格式图像的定位文件扩展名为 tw.tif, jpg 格式图像对应 jgw。② 采用 GeoTIFF 等专用的遥感图像格式,在文件头中保存定位信息。方法①目前使用的比较多,常用的 ArcGIS、MapInfo、

ERDAS 等地理信息系统软件和遥感图像处理软件均支持该方法,本文仅对该方法进行研究。

在定位文件中,按照 a 、 d 、 b 、 e 、 c 、 f 的次序记录了 6 个将图像坐标转换为地图坐标的参数。例如,一幅栅格图像的定位文件如下所示:

8.504 979 504 734 49	a
0.000 000 000 000 00	d
0.000 000 000 000 00	b
-8.504 979 504 734 49	e
728 610.370 965 644 700 00	c
3 848 213.298 466 135 500 00	f

图像中的像素按行列次序排列成矩阵形式,图像坐标系为左手坐标系,以像素为坐标单位,坐标原点在图像的左上角像素中心, x 轴指向行的方向, y 轴指向列的方向。如果某个像素的行号为 x_1 、列号为 y_1 ,则该像素的图像坐标可表示为 (x_1, y_1) 。在 AutoCAD 中,地图坐标系为右手坐标系,横轴 x 代表东方向,纵轴 y 代表北方向,单位一般为 m,可以用 (x_2, y_2) 表示像素的地图坐标。式(1)将像素的图像坐标转换为对应的地图坐标,这正是实现图像定位的依据。

$$\begin{cases} x_2 = ax_1 + by_1 + c \\ y_2 = dx_1 + ey_1 + f \end{cases} \quad (1)$$

显然,在上面的例子中, c 和 f 分别表示图像左上角像素中心对应的地图东坐标和北坐标; a 和 e 的大小分别代表像素在东方向和北方向的分辨率,即一个像素在这两个方向上代表的实际距离,一般情况下其数值相等,本例中一个像素对应约 8.5 m 的实地距离。由于图像坐标系与地图坐标系的 x 轴

收稿日期: 2011-01-21

作者简介: 龚有亮(1970—),男,内蒙古察右后旗人,硕士,副教授,主要研究方向为野外数字测图及卫星定位。

指向相同,而 y 轴指向相反,所以 a 为正而 e 为负。该例中的图像坐标和地图坐标仅存在平移缩放的关系,而没有旋转关系,所以 b 、 d 两参数为 0; 如果两者间存在旋转关系, b 、 d 两参数将不为 0, a 和 e 的大小也不再相等。据文献 [1] 测试, ERDAS、ENVI、ARCVIEW 等许多软件不支持旋转定位参数, 打开图像时并未将图像旋转, 而是将 b 、 d 两参数忽略, 造成图像定位出现偏差。经笔者测试, MapInfo 也是如此。

三、根据图像生成定位文件

如果一幅栅格图像已经在 AutoCAD 中正确定位(比如用南方公司的 CASS 软件进行过纠正), 但没有定位文件, 则可以通过编程来生成定位文件, 以方便将该图像在其他软件中使用。下面介绍如何使用 ObjectARX.NET 开发工具实现这一功能。

ObjectARX.NET 是 AutoCAD 最新提供的二次开发工具包^[2], 它是把面向 C++ 语言的 ObjectARX 开发包提供的类和函数进行封装而形成的托管类, 开发者可以使用 C# 或 VB.NET 语言调用这些类的功能, 为 AutoCAD 增加新的命令和功能。程序编译后生成扩展名为 dll 的动态库文件, 用“netload”命令加载后, 就可以像使用 AutoCAD 内部命令一样使用程序中自定义的命令。这种开发方式既继承了 ObjectARX 的强大功能, 又具有 .NET 开发简便、易用的特点, 是 AutoCAD 中极具应用前景的开发方式, 详细的开发方法可参考相关文献 [3-4]。

采用 C# 2008 进行编程, 获取图像定位参数的关键代码如下

```
[CommandMethod("WRITETFW")]
public static void WriteTfw()
{ Database db = HostApplicationServices.WorkingDatabase;
  Editor ed = Application.DocumentManager
    .MdiActiveDocument.Editor;
  //选择栅格图像
  PromptEntityOptions prEnt = new PromptEntityOptions
    ("\n 请选择栅格图像");
  prEnt.SetRejectMessage("\n 选择的不是栅格图像");
  prEnt.AddAllowedClass(typeof(RasterImage), true);
  PromptEntityResult prEntRes = ed.GetEntity(prEnt);
  if (prEntRes.Status != PromptStatus.OK) return;
  //取得像素坐标到模型空间坐标的变换矩阵
  Matrix3d PixelToModelTransform;
  using (Transaction trans = db.TransactionManager
    .StartTransaction())
  { ObjectId ImgId = prEntRes.ObjectId;
```

```
RasterImage imgObj = (RasterImage) trans
  .GetObject(ImgId, OpenMode.ForRead);
  PixelToModelTransform = imgObj
    .PixelToModelTransform;
  }
  ..... //将参数写入定位文件的代码省略
}
```

这段代码非常简短, 它定义了一个“WRITETFW”命令, 其功能是当用户在屏幕上用鼠标选择一幅图像后, 获取代表该图像的图像实体对象 (RasterImage 类型), 定位参数就保存在该对象的 PixelToModelTransform 属性中, 然后将其输出到文件中。

RasterImage 是表示栅格图像实体的类, 该类的每个实例代表一幅具体的图像, 通过该类的各种属性和方法, 可以获取或者设置图像的许多信息和特性, 包括图像文件的存储路径、图像的尺寸、显示比例、旋转角度、亮度、对比度等。这里用到的是 PixelToModelTransform 属性, 它属于 Matrix3d 类型, 其中存储了将图像坐标转换为 AutoCAD 模型空间坐标需要的各个参数。Matrix3d 是一种结构类型, 该类型变量中保存了一个如式 (2) 所示的三维坐标变换矩阵, 该类型广泛应用于 AutoCAD 中的各种坐标变换。由于图像定位属于二维坐标变换, 所以 PixelToModelTransform 属性中仅使用了其中的 6 个元素, 设像素的图像坐标为 (x_1, y_1) , 对应的模型空间坐标为 (x_2, y_2, z_2) , 实际上 AutoCAD 是按照式 (3) 将图像坐标转换为模型空间坐标, 从而实现图像定位, 要做的就是把这 6 个数按 a 、 d 、 b 、 e 、 c 、 f 写入定位文件中。

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} & t_0 \\ a_{10} & a_{11} & a_{12} & t_1 \\ a_{20} & a_{21} & a_{22} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & 0 & c \\ d & e & 0 & f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 0 \\ 1 \end{bmatrix} \quad (3)$$

四、根据定位文件插入图像

1. AutoCAD 图像功能的实现机制

在 AutoCAD 中通过编程插入图像比较复杂, 不仅要创建 RasterImage 类型的图像实体对象, 还要创建 RasterImageDef 类型的图像定义对象。这两个类之间的关系, 与块定义对象和块参照对象之间的关系非常相像^[5]。图像定义对象保存在名为 ISM_

RASTER_IMAGE_DICT 的特殊词典中,它是不可显示的,像块定义对象一样扮演着幕后角色,维护着与原始图像文件的连接,并执行显示和打印图像所要求的底层图像处理操作。图像实体对象代表显示在 AutoCAD 中的一幅图像,它像直线、圆、图块等实体对象一样,保存在块表的“模型空间”记录中。要插入图像,首先要创建图像定义对象,再创建图像实体对象,然后把它们关联起来,两个对象共同起作用,才能实现 AutoCAD 中的栅格图像功能。一个图像定义对象可以关联多个图像实体对象,也就是说同一幅图像在 AutoCAD 中可以有多个实例,其位置、尺寸等特性可以不同,但对应的是同一个图像定义对象,这多个图像实例的像素信息指向同一处内存空间。

2. 创建图像定义对象

下面的代码段用于创建图像定义对象:首先获取图像字典,若不存在则创建之,然后将其以写方式打开;接着创建图像定义对象,指定图像文件的实际路径并加载图像,再将图像定义对象加入到图像字典中。

```
ObjectId imgDictID = RasterImageDef
    . GetImageDictionary( db );
if ( imgDictID == ObjectId. Null ) { imgDictID =
    RasterImageDef. CreateImageDictionary( db ); }
DBDictionary imgDict = ( DBDictionary ) trans
    . GetObject( imgDictID , OpenMode. ForWrite );
RasterImageDef imgDef = new RasterImageDef( );
imgDef. SourceFileName = imageFileName; // 图像路径
imgDef. Load( );
ObjectId imgDefID = imgDict. SetAt( imgName , imgDef );
trans. AddNewlyCreatedDBObject( imgDef , true );
```

3. 创建图像实体对象

下一步创建图像实体对象,将图像定义对象的 ObjectId 属性值赋给图像实体对象的 ImageDefId 属性,使二者建立关联,并将图像实体对象附加到块表的“模型空间”记录中。代码如下

```
RasterImage imgRef = new RasterImage( );
imgRef. ImageDefId = imgDefID;
BlockTable bt = ( BlockTable ) tm. GetObject
    ( db. BlockTableId , OpenMode. ForRead , false );
BlockTableRecord btr = ( BlockTableRecord ) tm
    . GetObject( bt [BlockTableRecord. ModelSpace ] ,
    OpenMode. ForWrite , false );
btr. AppendEntity( imgRef );
trans. AddNewlyCreatedDBObject( imgRef , true );
imgRef. AssociateRasterDef( imgDef );
```

4. 图像的定位定向

至此,图像已经被成功插入到 AutoCAD 中,但其位置、缩放比例、方向等属性值均与期望的数值不同,下面将要解决图像的定位定向问题。要实现这一目标,必须正确设置图像实体对象的 PixelToModelTransform 属性值。现在已知图像的定位参数,因此容易构建这个 Matrix3d 类型的属性值,但该属性是只读的,不允许直接设置。经过深入研究,发现图像实体对象的另一个可读写的属性 Orientation 决定了 PixelToModelTransform 属性的值,进而决定了栅格图像的定位定向。Orientation 属性为 CoordinateSystem3d 类型,其构造函数为

```
public CoordinateSystem3d ( Point3d origin ,
    Vector3d xaxis , Vector3d yaxis );
```

其中,origin 是一个三维点,它决定了图像的插入点,即图像左下角的模型空间坐标;xaxis 是三维矢量,它的模决定了图像的宽度,它的方向表示图像宽的方向;yaxis 也是三维矢量,它的模决定了图像的高度,它的方向表示图像高的方向。因此,通过 Orientation 属性就可以确定图像的位置、缩放比例和旋转角,只要正确设置 Orientation 属性的值,AutoCAD 就会自动计算并设置 PixelToModelTransform 属性,从而实现图像的定位定向。

但是,根据定位参数直接计算 Orientation 属性值比较复杂,一个简单方法是根据 PixelToModelTransform 属性来反算 Orientation 属性。经过上述步骤插入的图像已经处于初始状态,具有缺省的位置、方向和尺寸,这时图像左下角位于 AutoCAD 坐标系的原点,旋转角为 0,宽度与图像中每行的像素个数相等,高度与每列的像素个数相等。先获取图像初始状态时的 PixelToModelTransform 属性,设为矩阵 A ,再根据定位参数构建图像正确定位时的 PixelToModelTransform 属性,设为矩阵 B ,则必然存在一个如式(4)所示的转换矩阵 M 可以将 A 转换为 B 。

$$B = MA, \quad M = BA^{-1} \quad (4)$$

求出转换矩阵 M ,用它对图像初始状态时的 Orientation 属性值进行转换(实际上是对它的 origin、xaxis、yaxis 三个属性值分别进行转换),计算出图像正确定位时的 Orientation 属性值,然后正确设置图像实体对象的该属性即可实现图像的正确定位。实现这一过程的代码如下所示

```
double [] matElements = new double [16]
    { a b 0 c d e 0 f 0 0 1 0 0 0 0 1 }; // 定位参数
```

(下转第 22 页)

可以满足导航的需求,但在遇到拐弯或载体运动为非匀速直线运动时,只采用预报值无法满足导航的需要。故该模型只能解决短时间的 GPS 信号缺失问题。

参考文献:

- [1] 李征航,黄劲松. GPS 测量与数据处理[M]. 武汉: 武汉大学出版社, 2005.
- [2] 陈俊勇. 中国国家高精度 GPS 大地控制网的建立[J]. 测绘通报, 1998(8): 1-2.
- [3] 施一民. 现代大地控制测量[M]. 2版. 北京: 测绘出版社, 2008.
- [4] 同济大学, 武汉测绘科技大学. 控制测量学(下)[M]. 北京: 测绘出版社, 1986.
- [5] 杨元喜, 高为广. 两种渐消滤波与自适应抗差滤波的综合比较分析[J]. 武汉大学学报: 信息科学版, 2006, 31(11): 44-46.
- [6] 刘大杰, 陶本藻. 实用测量数据处理方法[M]. 北京: 测绘出版社, 2000.
- [7] YANG Yuanxi, HE Haibo, XU Guochang. Adaptively Robust Filtering for Kinematic Geodetic Positioning[J]. Journal of Geodesy, 2001, 75(2): 109-116.
- [8] 胡国荣, 欧吉坤. 改进的高动态 GPS 定位自适应卡尔曼滤波方法[J]. 测绘学报, 1999, 28(4): 10-14.
- [9] 陈晓晖, 郭美凤, 毛献辉, 等. 自适应卡尔曼滤波在移动卫星通讯系统中的应用[J]. 中国惯性技术学报, 2001, 9(2): 24-27.
- [10] 杨元喜, 唐颖哲, 李庆田, 等. 用于 GIS 道路信息修测的动态 GPS 自适应滤波试验[J]. 测绘科学, 2003, 28(12): 9-11.

(上接第 15 页)

```
Matrix3d mat1 = new Matrix3d( matElements );
Matrix3d mat = mat1 * imgRef
    . PixelToModelTransform. Inverse( );
Point3d Origin1 = mat * imgRef. Orientation. Origin;
Vector3d Xaxis1 = mat * imgRef. Orientation. Xaxis;
Vector3d Yaxis1 = mat * imgRef. Orientation. Yaxis;
imgRef. Orientation = new CoordinateSystem3d
    ( Origin1, Xaxis1, Yaxis1 );
```

五、试验与结论

笔者在 AutoCAD 2008 环境下,按照本文所述方法编程实现了根据定位文件插入图像以及根据图像生成定位文件的功能。并且将同样的图像分别在 AutoCAD、ArcGIS、MapInfo 中插入,在三种软件中分别量取相同像素点的地图坐标并相互比较,结果完全一致,证明图像定位准确无误。同时,该方法

可对具有任意旋转角、纵向和横向缩放比例不一致的图像进行定位,具有良好的适应性和灵活性。

参考文献:

- [1] 王强,束炯,张晓沪. 一种遥感图像的坐标转换方法[J]. 测绘科学, 2006, 31(4): 137-139.
- [2] 李阳东,董小华,刘妙龙. 几个基于 ObjectARX. net 的 GIS 功能算法[J]. 测绘与空间地理信息, 2009, 32(1): 11-14.
- [3] 曾洪飞,张帆,卢择临. AutoCAD VBA & VB. NET 开发基础与实例教程[M]. 北京: 中国电力出版社, 2008.
- [4] 秦洪现,崔惠岚,孙剑,等. Autodesk 系列产品开发培训教程[M]. 北京: 化学工业出版社, 2008.
- [5] Autodesk. ObjectARX for AutoCAD(Managed Class Reference) [EB/OL]. 2010-5-20 [2010-7-20]. [http://www.objectarx.net/downloads/arxref\(.net\).zip](http://www.objectarx.net/downloads/arxref(.net).zip).

《ArcGIS 制图和空间分析基础实验教程》出版

[本刊讯] 由北京建筑工程学院王文宇、杜明义编著的《ArcGIS 制图和空间分析基础实验教程》近日由测绘出版社出版。

该书以 ArcGIS 软件为例,引导读者由浅入深地掌握 GIS 软件操作技巧。通过基础实验,读者将首先熟悉空间数据库的建立和制图等基本操作,然后逐步过渡到空间分析与建模等核心技术,最后通过真实案例实践掌握 GIS 在实际地学中的应用。该书共由 18 个独立实验组成,读者可以按照书中指导的步骤,有效地完成实验,并了解 GIS 软件的基本功能和初步掌握 GIS 的基础空间分析能力(书中的实验数据可在网上下载)。该书是一本 GIS 软件操作的入门指导书,适合地理信息系统和相关专业的高职高专和本科学学生使用。

该书为 16 开本,定价 19.80 元。

(本刊编辑部)