Geomatics Science and Technology 测绘科学技术, 2015, 3, 34-41
Published Online April 2015 in Hans. http://www.hanspub.org/journal/gst
http://dx.doi.org/10.12677/gst.2015.32006

Hans汉斯

A Quick Tile Cache Generation Method Based on Dynamic Projection and Scan-Line Cropping

Zhao Yan, Nan Wang

School of Remote Sensing and Information Engineering, Wuhan University, Wuhan Hubei
Email: whuhenry@whu.edu.cn

Received: Apr. 1st, 2015; accepted: Apr. 17th, 2015; published: Apr. 22nd, 2015

Abstract

Tile map cache is a kind of service cache technique for quick accessing static map. It is developed during the progress of Web GIS. In recent years, it is widely used in 3D terrain display as the fundamental data organization method. In the progress of tile map production, original image data are usually different in data format, projection, resolution and so on because of various data sources. Meanwhile, there are other restrictions such as image cropping by area of interest because of application localization or secrecy. So the traditional map tile generation begins with image re-projection, cropping and mosaic to produce united semi-finished image, and then puts it into map tiling software. This method is inefficient and takes lots of storage. Based on the method, this article suggests a dynamic projection using linear polynomial numerical projection and image quick cropping algorithm based on scan-line filling. The experimental results show that the method can realize automatic tile generation in one-time processing. The time it used is nearly the same as traditional methods which only deal with united semi-finished image. The algorithm can finish the map tiling without generating semi-finished image. As a result, it can improve the automation and reduce the processing time.

Keywords

Tile Pyramid, Rapid Tile Generate, Dynamic Re-Project, Scan-Line Cropping

一种基于动态投影和扫描线裁剪的快速瓦片缓存生成方法

闫　钊，王　楠

武汉大学遥感信息工程学院，湖北 武汉
Email: whuhenry@whu.edu.cn

收稿日期：2015年4月1日；录用日期：2015年4月17日；发布日期：2015年4月22日

摘 要

瓦片地图缓存技术是在网络GIS发展过程中产生的一种能够实现静态数据快速浏览的服务器缓存技术，同时近年来也被广泛用于三维地形显示中作为底层数据组织方式。在瓦片地图生产过程中，由于原始影像来源广泛，通常数据格式、投影方式、影像分辨率等均不一致，同时在切片过程中会存在根据地域范围或保密要求等因素需要对影像进行裁剪的等其他约束条件，在传统的切图流程中，会首先在遥感影像处理软件中对影像进行重投影、裁剪、拼接等操作，生成统一格式的中间影像，然后再输入到切图系统中，这种方式不仅效率低下，同时占用大量存储空间。本文在传统切图方法基础上，提出了在切图过程中逐瓦片使用一次多项式进行数值重投影实现的动态投影方法，以及基于扫描线填充算法的影像快速裁剪算法。通过实验对比分析，在多约束条件下使用本文算法能够实现一次性自动化切图，切图效率同无约束条件下切图效率相当，能够在基本不影响切图效率的同时完成瓦片金字塔构建，同时省去了影像预处理过程，极大的提升了处理自动化程度，降低了处理时间。

关键词

瓦片金字塔，快速切图，动态投影，扫描线裁剪

1. 引言

随着摄影测量和遥感硬件技术的快速发展，影像数据的质量和数量快速提高，数据量呈几何级数增长，同时随着计算机网络的快速发展，几乎所有的影像数据都需要通过网络服务进行分发，影像服务面临着数据密集型、计算密集型和时空数据并发访问的挑战[1] [2]。如何能够高效的管理海量影像数据，快速浏览和检索特性的数据成为近年来研究和工程实践中的热点问题。

瓦片地图技术是随着网络 GIS 发展而产生的一种能够快速浏览地图数据的技术，通过将单幅大文件影像数据预缓存为小块的压缩影像数据，提高了网络服务时的影像检索速度并降低了网络传输时间。近年来，随着数字地球的快速发展，瓦片地图同时也广泛用于三维地形数据组织中，用来存储海量高程数据和影像数据。然而从多源遥感影像生成瓦片金字塔首先要解决影像投影问题，由于影像需要逐点计算投影后坐标，但是考虑到影像点数量巨大(例如一幅 256*256 大小的瓦片就需要计算 65,536 个点，而瓦片数量通常为几十万，上百万甚至上千万张)，使用传统的矢量投影中使用的严密投影计算较慢，严重影响处理效率；此外由于原始影像成像质量问题或保密等原因，影像切片处理时经常需要划定感兴趣区域(Area of Interest, AOI)并进行裁剪。传统处理方法一般采用分步处理的方法来满足这一需求，即首先对影像进行投影变换，然后进行影像裁剪和拼接，最后切片输出。这种方法不仅处理速度低下，同时会浪费大量的存储空间用于存储中间临时数据。因此本文重点从影像切片过程中的动态投影和动态裁剪两方面入手，设计并实现快速处理算法，提高瓦片处理效率。

2. 影像动态投影

影像投影是指按照一定的数学法则，将地球椭球面上的点坐标转换到平面上。在瓦片金字塔生成过程中由于目标金字塔的唯一性，因此要求所有的输入影像均需要首先进行投影变换才能进行瓦片切片。

投影转换方法一般有如下三种：直接变换法、反解变换法和数值变换法[3]。其中前两种方法需要原始坐标系和目标坐标系的数学描述都已知，并且需要知道详细的描述参数(例如参考椭球参数、投影变换中央经线参数等)。而数值变换方法无需知道投影方法的数学描述，只需要知道变换公式和相关参数即可。

在投影变换精度和变换速度方面，由于直接变换法和反解变换法准确知道投影的数学模型，因此能够做到准确变换，但通常来说投影数学模型都比较复杂，变换过程中需要进行大量的开方，乘方和三角函数等浮点运算，因此计算速度相对较慢。而数值变换法的精度依赖于计算参数的个数和变量乘方次数，一般来说计算参数越多、变量乘方次数越高，计算结果越精确[4]，但是同时计算时间也会越长，因此常用的数值变换方法通常为一次变换、二次变换和三次变换。同直接变换法和反解变换法相比，数值变换法的计算速度通常能够提升一个甚至两个数量级，因此通常对于影像投影变换这一类需要海量计算的投影变换需求，通常使用数值变换法进行投影变换。

2.1. 影像快速投影变换算法设计

本文针对瓦片生成过程中中间数据的特点，设计了一种快速投影变换方法，在切片过程中，每一个单独瓦片使用一次数值变化法进行投影变换计算，而由于每张瓦片相对代表的实际地表面积很小，使用数值变换时整体变形很小，基本能够保证投影变换的精度。具体流程如下：

1) 选择控制点，由于使用一次数值投影变换法，需要至少在影像上选择 3 个控制点，考虑到提高变换精度以及选点的方便，使用每张瓦片的 4 个角点作为控制点，对控制点使用 PROJ.4 库进行精密的投影变换计算，得到在目标投影坐标系下控制点的精确坐标。

2) 解算变换参数。根据一次数值变换定义[5]，从原始坐标系的(x, y)点变换到目标投影坐标系的(x', y')点，需要满足如下公式：

$$\begin{cases} x' = a_{11}x + a_{12}y + a_{13} \\ y' = a_{21}x + a_{22}y + a_{23} \end{cases} \tag{1}$$

其中 a_{11}~a_{22} 均为一次变换参数，这样在解算参数时可以将它们作为未知数，将坐标值作为已知数，可以构建如下矩阵方程

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ x_3 & y_3 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_3 & y_3 & 1 \\ x_4 & y_4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_4 & y_4 & 1 \end{bmatrix} \begin{bmatrix} a_{11} \\ a_{12} \\ a_{13} \\ a_{21} \\ a_{22} \\ a_{23} \end{bmatrix} = \begin{bmatrix} x_1' \\ y_1' \\ x_2' \\ y_2' \\ x_3' \\ y_3' \\ x_4' \\ y_4' \end{bmatrix} \tag{2}$$

其中 x_1~x_4 和 y_1~y_4 代表瓦片四角点坐标值，$x_1' \sim x_4'$ 以及 $y_1' \sim y_4'$ 代表四个角点对应的影像坐标值。由于该方程参数多于未知数格式，因此采用最小二乘法求解该方程，并最终得到数值投影变换的一次变换参数 a_{11}~a_{23}。

3) 变换合理性检验。在某些特殊情况下，例如低层级瓦片代表的实际范围过大，或者投影坐标系与原始坐标系参数差距过大时，使用一次投影变换得到的结果会产生很大的偏差，因此，需要增加正确性检验。本文使用瓦片四条边界中点和瓦片中心点作为检查点，将检查点坐标带入公式(1)中，根据步骤(2)计算得到的参数计算出检查点的坐标值；同时使用 PROJ.4 库计算检查点投影后的真实坐标值，两者相比

较如果误差在 0.1 个像素以内，满足数字制图精度要求，则认为可以使用数值计算方法代替精确计算方法，并进行影像投影变换，否则需要对整个瓦片进行精确的投影变换完成瓦片切割。

2.2. 算法的适用性和效率分析

为了验证上述算法的适用性和可靠性，本文使用如下金字塔组织方式：使用 WGS84 坐标系(EPSG：4326)以经纬度为单位，经度范围为从西至东[−180, 180]，纬度范围为从北至南[90, −90]，顶层第 0 级瓦片为2张正方形瓦片，瓦片边长为256像素，经度范围分别为[−180, 0]以及[0, 180]，纬度范围均为[90, −90]，而后向下每一层分别由上一层瓦片等分四个正方形。选择北京(116.390058E, 39.909565N)、武汉(114.331947E, 30.536259N)和广州(113.277602E, 23.127464N)三个地点作为测试区域中心点，计算该点所在瓦片金字塔每一层瓦片使用逐点精确投影方法和本文设计快速投影方法两者分别计算从 WGS84 坐标系转换到中国2000投影坐标系(根据分带规则，武汉和广州为38号带[EPSG：4547]，北京为39号带[EPSG：4548])。以精确投影方法为基准，计算快速投影方法与其在精度和纬度方向的偏差。得到的结果如图 1 所示。

图中横轴表示瓦片所在层级，均从第 7 级瓦片开始计算，纵轴表示每张瓦片中所有点坐标计算误差的最大值相对于该层瓦片的一个像素的大小。从图中可以看出，随着瓦片层级的增大，误差在迅速减小。同时根据数字地图制图要求(定位误差应控制在 0.1 个像素内)，三个实验地区均从第 10 级开始，投影计算误差小于 0.1 个像素大小。而第 10 级对应的地面分辨率约为 75 米，因此一般来说对于影像分辨率小于 75 米的影像均可以使用快速投影方法进行瓦片重投影计算，而目前常用的卫星遥感影像和航空摄影相片，其影像分辨率大都高于这一要求，所以对于大多数影像均可以使用该方法进行瓦片快速投影计算。

在投影计算效率方面，同样使用上述瓦片金字塔结构和实验区域，选择第 17 级瓦片作为实验对象，使用 PROJ.4 库(4.8.0 版本)进行标准投影变换计算，对该张瓦片进行 10,000 次投影变换计算，测试使用的硬件条件为 Intel Core i7-4700MQ 四核处理器，8GBDDR3L 内存和 1TB 笔记本硬盘(5400 转/秒)，得到结果如表 1 所示。

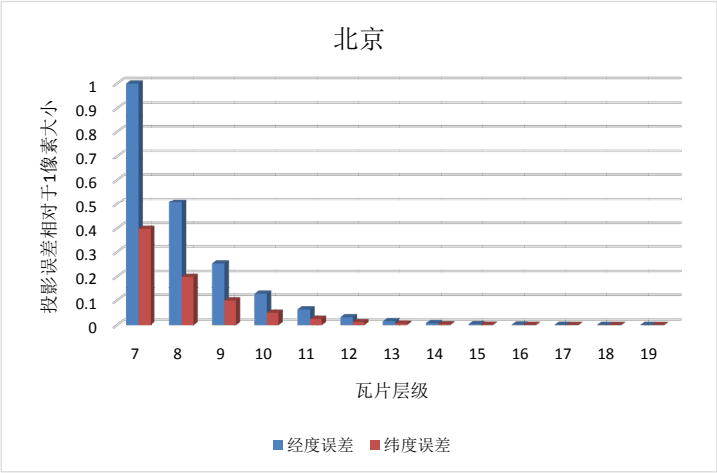
从表格可以看出如果完全使用精确投影变换计算，平均每张瓦片需要 9 ms 左右才能完成投影变换，也就是每秒最多处理 110 张瓦片，这极大的影像的瓦片生成效率。而使用本文提出的快速投影变换算法，可以将处理速度提高 2 个数量级，使单张图片的投影处理时间可以忽略不计。

3. 影像快速裁剪

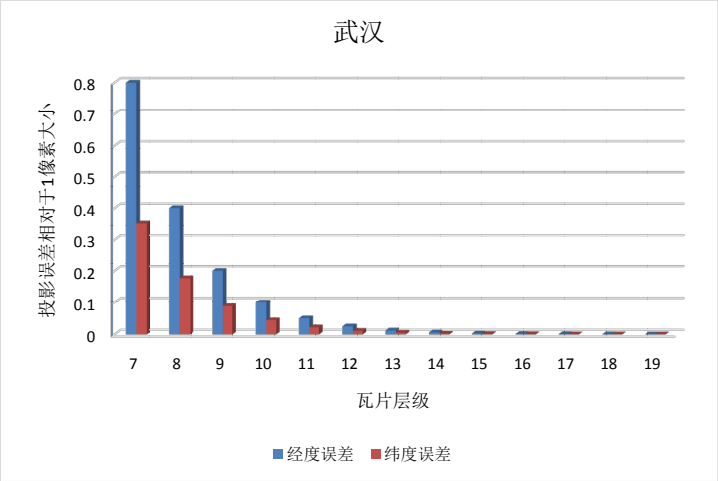
原始栅格影像，特别是卫星遥感影像数据，数据边缘通常有不规则的无数据黑色边缘，用户在使用该数据时通常自行勾画出有效数据范围对影像进行裁剪后使用；此外用户使用的数据通常要求是某个行政区划范围内或者其他类型的自定义区域内。在传统的瓦片数据生产过程中通常做法为首先使用给定的 AOI 对影像进行裁剪，生成只包含有效数据的中间影像，再对中间影像进行切片处理。这样的处理过程会生成大量的中间影像数据，同时由于中间数据需要存储在硬盘上，大量的数据 I/O 操作会极大的降低瓦片生成速度。因此本文借鉴扫描线填充算法的思想[6]-[8]，提出了一种在栅格数据切片过程中同步进行影像裁剪的方法，能够实现在瓦片生成的同时对影像进行裁剪处理，并保证瓦片生成速度基本不变。

3.1. 基于扫描线填充算法的影像快速裁剪方法

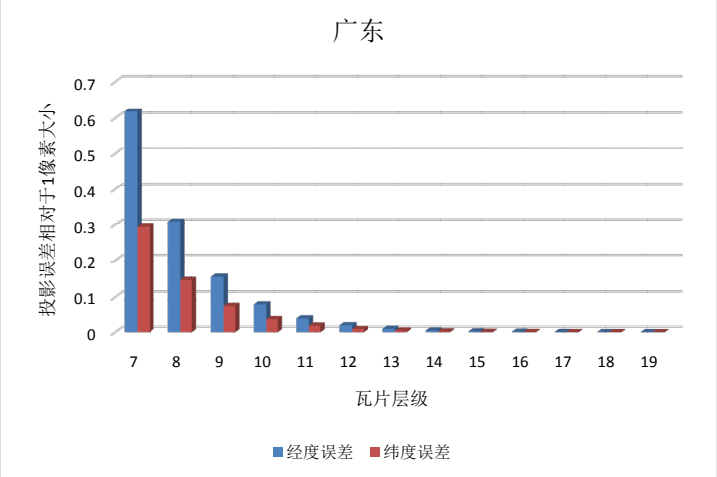
我们定义 AOI 的表达方式为使用简单多边形表达，多边形不能自交叉、不能存在岛、洞等情况。首先将 AOI 数据投影到瓦片金字塔使用的投影坐标系下。然后计算出原始栅格数据覆盖的瓦片范围，x 轴方向从第 x_0 列到第 x_1 列，y 轴方向从第 y_0 行到第 y_1 行，同时根据瓦片切片时定义的每个瓦片的边长 l (单位为像素)，这样可以将整个瓦片区域看成为一张长为 $(x_1 - x_0 + 1)\times l$，宽为 $(y_1 - y_0 + 1)\times l$ 的图片，这样



(a)



(b)



(c)

Figure 1. Dynamic projection error of each level in latitude and longitude direction
图 1. 动态投影方法在不同层级经纬度方向误差

Table 1. Time cost comparison between strict projection and dynamic projection (in ms)
表 1. 严格投影变换和动态投影变换所需时间对比(单位：毫秒)

投影计算方法	北京	武汉	广州
Proj.4	89,058	97,419	89,506
动态投影变换	90	107	87

可以使用扫描线填充的法确定 AOI 覆盖的整个像素范围，逐行记录每一行 AOI 覆盖的边界值，每一行记录包括 $2n$ 个数值，这些数值从小到大排序，每一对表示改行上的一个线段起止点，在该线段内的所有点都在 AOI 内，即所有这些点都需要从原始影像中进行采样获取数据。

具体的如图 2 所示，AOI 共覆盖 9 行像素区域，最后一行没有覆盖的区域记录中没有任何数值，第 (y_0+1) 行和第 (y_0+8) 行分别有两个线段，因此需要记录 4 个值，以第 (y_0+1) 行为例，第一个线段为 $[x_0+2, x_0+5]$，第二个线段为$[x_0+7, x_0+8]$，在这两个线段内的像素(图中为深色方块)为有数据区域，需要从原始数据中采样填充颜色，而其他白色方块所表示的像素为无数据区域，可以直接跳过。需要注意的是，如果某个线段长度只有 1，即起止点为同一点，也需要记录两个数值，使程序能够统一处理(如第 y_0 行)。

3.2. 实验和结果分析

为了验证算法的实际效率，选择一幅汉中地区的三波段影像数据和对应的 AOI 数据作为测试数据(如图 3 所示)，影像大小为 44,288*31,488，文件大小为 3.89 GB，AOI 数据为一简单多边形，共包含 23 个顶点。使用软件分别测试原始影像直接生成瓦片和使用 AOI 裁剪生成瓦片两种情况下瓦片生成效率，测试使用的硬件条件为 Intel Core i7-4700MQ 四核处理器，8 GB DDR3L 内存和 1 TB 笔记本硬盘(5400 转/秒)，实验结果如表 2 所示。

从结果可以看出，由于使用 AOI 裁剪，整体需要输出瓦片的影像数量减小，因此瓦片生成时间比原始影像直接生成瓦片要少；同时由于使用快速裁剪算法，在瓦片生成过程中实时进行影像裁剪，瓦片生成效率(每秒生成瓦片数量)和直接生成瓦片基本一致。这样既节省了影像预处理工作中影像裁剪所需要的大量的计算和磁盘 I/O 时间，同时由于不需要存储中间临时数据，节省了大量的存储空间。

对于切片效率来说，通常不会随 AOI 的复杂而导致效率降低，究其原因，本算法只需要在切片初始化时对每张原始栅格数据进行一次 AOI 裁剪计算，并在内存中保存计算结果，在切片计算过程中，对于每一张瓦片，逐点判断是否需要从原始栅格数据中采样填充。这样对于瓦片生成过程只需要增加一个简单的判断，几乎不会影响切片计算速度，而预处理计算过程等价于对一个简单多边形进行扫描线填充算法，其计算复杂度为 $O(n)$，其中 n 代表影响覆盖瓦片的像素行数，一般来说这个行数不大于原始影像行数，而原始影像行数最大一般在 10^6~10^7 数量级，按照现在一般计算机的计算性能，计算时间通常不足 1 s，而对于一幅 1 G 左右大小的栅格数据，切片所用时间为 30 s 左右，可以看出该裁剪方法对瓦片生成效率影响微小，并且原始影像越大影响越微小。

4. 结论和展望

影像切片作为网络地图服务的必备准备工作，随着数据的飞速增长和需求的不断变化面临着越来越大的挑战，本文从传统切图服务的工作流程出发，将影像的投影变换和 AOI 裁剪两项功能合并到切图过程中，设计并实现了相关算法，通过测试，能够在进行瓦片动态投影和 AOI 裁剪的同时达到生成单层级瓦片 150 张/秒，生成瓦片金字塔 70 张/秒的生成速度。极大的满足紧急情况下瓦片切分任务和日常瓦片金字塔更新维护任务。

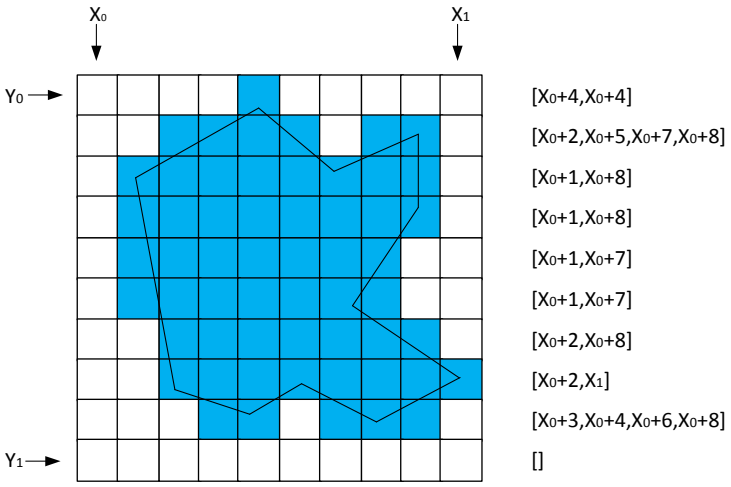


Figure 2. Storage method of AOI cropping edge point
图 2. AOI 裁剪边界点存储方式

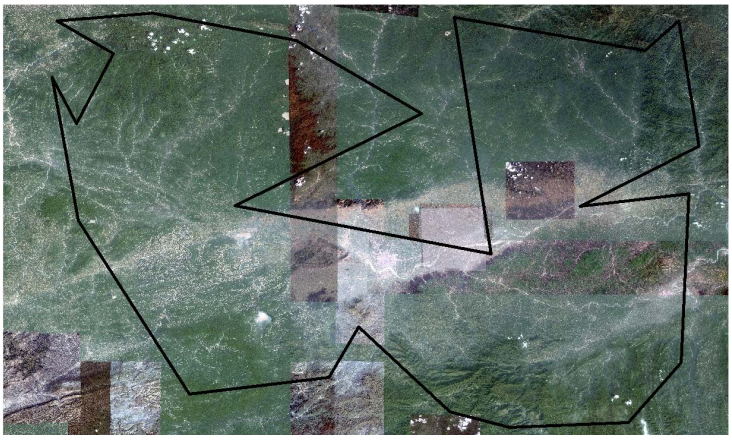


Figure 3. Raster data and AOI of Han Zhong area used in experiment
图 3. 汉中地区测试栅格数据和 AOI

Table 2. Tile cache generation time with and without AOI
表 2. 使用 AOI 裁剪与不裁剪生成瓦片时间对比

投影计算方法	生成瓦片数量	时间(秒)
原始影像	24,062	68
使用 AOI 裁剪	14,618	41

但同时在应用过程中也发现，目前设计并实现的单机瓦片切图存在系统硬盘 I/O 瓶颈以及瓦片切图服务器失效后无法恢复等问题，同时单机瓦片存储容量也有一定限制。因此在未来的研究中可以借助云计算相关技术(MapReduce 和 MPI 等)以及分布式存储技术(GFS，DFS 和 NoSQL 等)实现分布式切图系统，并解决瓦片拼接，更新等问题，实现真正的高性能、高可用性、易用和动态可扩展的切图系统，为空间基础设施建设服务。

基金项目

民用航天“十二•五”预先研究项目基金(No. 2013669-7)资助。

参考文献 (References)

[1] 饶庆云, 丁晶晶, 苏乐乐, 等 (2013) 基于云计算的分布式切图服务设计与实现. *测绘与空间地理信息*, **S1**, 29-35.

[2] Qin, C.-Z., Zhan, L.-J. and Zhu, A.X. (2014) How to apply the geospatial data abstraction library (GDAL) properly to parallel geospatial raster I/O. *Transactions in GIS*, **18**, 950-957.

[3] 刘庆元, 孟俊贞 (2009) 双线性插值近似网格的栅格数据投影变换. *测绘工程*, **5**, 15-17+21.

[4] 刘宏林, 吕晓华, 江南, 等 (2011) 影响地图投影多项式变换精度的若干因素分析. *测绘科学技术学报*, **2**, 108-112.

[5] 吕晓华, 刘宏林 (2002) 地图投影数值变换方法综合评述. *测绘学院学报*, **2**, 150-153.

[6] 潘俊 (2005) 立体正射影像无缝镶嵌技术研究. 硕士论文, 武汉大学, 武汉.

[7] 潘俊, 王密, 李琪 (2006) 基于扫描线填充的快速镶嵌算法. *测绘信息与工程*, **5**, 8-9.

[8] Anon (1985) Graphics area fill algorithm. *IBM Technical Disclosure Bulletin*, **27**, 6994-6996.