Google 地图算法研究及实现

崔金红[1] 王 旭[2]

(对外经济贸易大学信息学院 北京 100029)[1] (贝尔实验室基础科学研究院 北京 100080)[2]

摘 要 Google Map、Google Earth 以其丰富的地图和黄页资源,特别是覆盖全球的卫星地图而著称。本文系统地研究了 Google 地图的算法及实现,并给出了一些 Google 地图读取的图例,以帮助用户理解 Google 地图的机制,更有效地将 Google 地图集成到应用系统中。

关键词 Google 地图,地图投影,墨卡托投影,GIS

Research on Google Map Algorithm and Implementation

CUI Jin-Hong[1] WANG Xu[2]

(School of Information Technology & Management Engineering, University of International Business and Economics, Beijing 100029)[1]

(Bell Labs Research China, Beijing 100080)[2]

Abstract Google Map and Google Earth are famous for their rich map and yellow page resource. The principal and implementation of Google Map are studied in this paper, and some samples generated by the demo system are given. This paper helps better understand the mechanism of Google Map, and integrate Google Map function into application system.

Keywords Google map, Map projection, Mercator projection, GIS

1 引言

Google Map、Google Earth 自推出以来,以其强大的功能、丰富的地图和黄页资源,特别是它覆盖全球的卫星地图,吸引了越来越多的用户。尽管 Google 也提供相应的 API,以使用户在自己的系统中集成 Google 地图的功能,但其提供的 API 在某些情况下却不能直接使用,如一些不支持 Ajax 的浏览器、某些 IPTV 机顶盒、手持终端以及传统 C-S 架构的应用等等。

通过对 Google Map 的研究,我们发现 Google Map 采用墨卡托(Mercator)投影,也就是等角圆柱投影,其地图均是由 256 * 256 的小图块拼接而成,而且这些图块的 URL 都是不变的。本文对墨卡托投影及其在 Google Map 的应用、算法作些探讨,基于这些算法构造了一个 Google 地图读取系统,并给出了一些 Google 地图读取的图例。

2 Google Map 数学模型

所谓地图投影,就是在平面上建立与地球曲面上相对应的经纬网的方法。地图投影的拟定和计算,一般均假定地球表面为旋转椭球面,并称其为地球椭球面或参考椭球面。

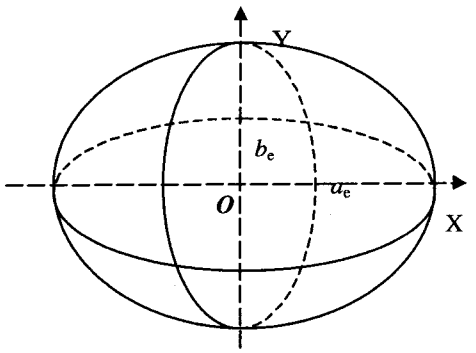


图 1 地球椭球体示意图

如图 1 所示,地球椭球体的形状和大小是由其长半径 a_e(赤道半径)和短半径 b_e(极轴半径)决定,通常取 a_e 值为 6378137m,b_e 值为 6356752.3m。

另外,用于描述地球椭球形状的参数还有椭圆扁率 f_e、第一偏心率 e_1 和第二偏心率 e_2,其中

$$\left.\begin{aligned} f_e&=\frac{a_e-b_e}{a_e}\approx 0.00335281\ 29\\ e_1&=\sqrt{\frac{a_e^2-b_e^2}{a_e^2}}\approx 0.08181921\ 8\\ e_2&=\sqrt{\frac{a_e^2-b_e^2}{b_e^2}}\approx 0.08209446\ 54 \end{aligned}\right\}$$

2.1 墨卡托投影

墨卡托投影是等角圆柱投影,如以赤道作 X 轴,根据等角条件推算出的投影公式为

$$\left.\begin{aligned} x&=a_e\cdot\lambda\\ y&=a_e\cdot\ln\tan\left(45^\circ+\frac{\varphi}{2}\right)\cdot\left(\frac{1-e_1\sin\varphi}{1+e_1\sin\varphi}\right)^{\frac{e_1}{2}} \end{aligned}\right\}$$

其中 λ 为经度,φ 为纬度。

实际计算中,可将上述公式简化,将地球作为球体来处理,球体的半径取地球几何平均半径 $R_e=6371004$m,则

$$\left.\begin{aligned} x&=R_e\cdot\lambda\\ y&=R_e\cdot\ln\tan\left(45^\circ+\frac{\varphi}{2}\right) \end{aligned}\right\}$$

2.2 Google 地图投影

Google 地图分为三大类,即普通地图、卫星地图以及合成地图,其中合成地图由卫星地图和底色透明的普通地图叠加而成。它们都是由 256×256 大小的 png 图片拼接而成,每块图片的 URL 格式为:

普通地图

http://mt.google.com/mt? n=404&v=w2.33&x=?

崔金红 博士,副教授,主要从事信息系统、工作流等研究;**王 旭** 博士,高级研究员,主要从事计算机图形学、科学可视化、计算机通信及应用、IMS/NGN 及 3G 应用等研究。

&y=？ &zoom=？

卫星地图

http://kh.google.com/kh? n=404&v=13&t=？

底色透明普通地图

http://mt.google.com/mt? n=404&v=w2t.34&x=？ &y=？ &zoom=？

参数 v 表示图源数据版本，参数 zoom 为缩放倍数，其取值范围为 0～17，x 表示经度方向图片编号，y 表示纬度方向图片编号，x、y 的取值范围则为 $0\sim 2^{17-zoom}-1$，参数 t 是“qrst”4 个字符排列而成的字符串，表示卫星地图图片编号。

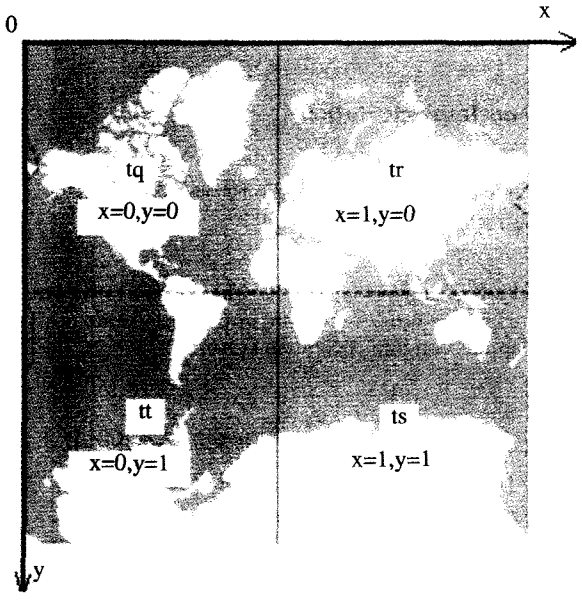


图 2 Google 图块分布规则

如图 2 所示，Google 地图在 zoom 等于 17 时，全球就为一个 256 * 256 的图片，它的中心经纬度为(0,0)，t 值为“t”，x 值为 0，y 值为 0；当 zoom=16 时裂化为 4 块，每块的编号为：左上 t=“tq”，x=0，y=0；右上 t=“tr”，x=1，y=0；右下 t=“ts”，x=0，y=1；左下 t=“tt”，x=1，y=1。依此类推，每放大一倍，每一小块都分裂为四块，卫星图片从左上到右下顺时针按“qrst”编号，分裂后的图块编码为分裂前的编号加上小块的编号。

为获取某经纬度图片的 URL，就需要把经纬度转化为 x、y 坐标。Google 地图坐标的计算公式为

$$\left.\begin{aligned} x&=(\mathrm{int})\frac{(180+\lambda)\cdot 2^{17-zoom}}{360}\\ y&=(\mathrm{int})\left(\left(\frac{(\ln\tan(46^\circ-\frac{\varphi}{2})}{2\cdot PI}+0.5\right)\cdot 2^{17-zoom}\right)\end{aligned}\right\}$$

其中 PI 为圆周率；λ 为东经，取值范围为 −180～+180 度；φ 为北纬。

图 3 所示为 $\ln\tan(45^\circ-\frac{\varphi}{2})$ 的函数曲线图。从该图中可以看到，当 $\varphi\approx\pm 85.051128$ 时，函数值为 ±PI，Google 坐标 y 值将达到极值 0 和 $2^{17-zoom}$，因而上述坐标计算公式应满足 $-85.051128<\varphi<85.051128$。那么，对于 $\varphi\leqslant -85.051128$ 及 $\varphi>85.051128$ 及如何处理呢？由于在两极附近，Google 不提供高放大倍数的图片。实际处理中，当 $\varphi\leqslant -85.051128$ 时，可取 y 值为 $2^{17-zoom}-1$；当 $\varphi>85.051128$ 时，取 y 值为 0 即可。

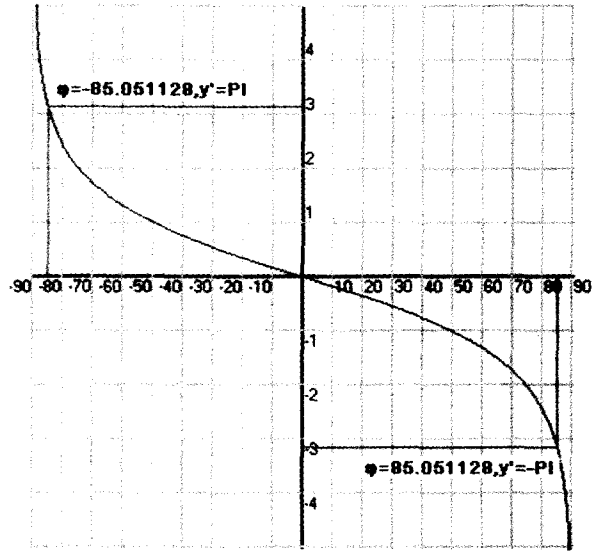


图 3 墨卡托函数曲线图

反之，图片(x,y)的左上角和右下角经纬度分别是

$$\left.\begin{aligned}\lambda_t&=\frac{360^\circ}{2^{17-zoom}}\cdot x-180^\circ\\ \varphi_t&=(45^\circ-\mathrm{atan}(e^{\left(\frac{y}{2^{17-zoom}}-0.5\right)\cdot 2\cdot PI}))\cdot 2\end{aligned}\right\}$$

$$\left.\begin{aligned}\lambda_b&=\frac{360}{2^{17-zoom}}\cdot(x+1)-180\\ \varphi_b&=(45^\circ-\mathrm{atan}(e^{\left(\frac{y+1}{2^{17-zoom}}-0.5\right)\cdot 2\cdot PI}))\cdot 2\end{aligned}\right\}$$

其中 e 为自然对数底数。

特殊地，当 $y=0$ 时，图片左上角纬度取值为 90 度；当 $y=2^{17-zoom}-1$ 时，图片右下角坐标取值为 −90 度。

2.3 距离计算公式

假定地球为一球体，取其平均半径 $Re=6371004$，则地球表面两点 $P_o(\lambda_o,\varphi_o)$、$P_d(\lambda_d,\varphi_d)$ 间的距离为

$$Re\cdot\mathrm{acos}(\cos\lambda_o\cdot\cos\varphi_o\cdot\cos\lambda_d\cdot\cos\varphi_d+\sin\lambda_o\cdot\cos\varphi_o\cdot\sin\lambda_d\cdot\cos\varphi_d+\sin\varphi_o\cdot\sin\varphi_d)$$

3 系统设计

假定用户需要获得中心点为 $P(\lambda,\varphi)$、长度为 W 像素、宽度为 H 像素、观测半径为 r 米的地图图像。

3.1 Zoom 值确定

为了使用 Google 地图，首先必须确定 zoom 值。可将不同 zoom 值所对应的(m/像素)值存入一个 Hash 表中，然后根据用户要求的图幅和观测半径计算其对应的(m/像素)值，并与前面保存的表进行比对，取最接近的 zoom 值即可。

3.2 拼图算法

为了返回用户需要的地图，首先根据上文介绍的公式计算出地图中心点 $P(\lambda,\varphi)$ 所在图块的 x、y 坐标值及 t 值；然后根据图幅及中心点坐标计算出所需图块的数量，以及各图块所对应的 x、y 坐标值及 t 值，进而得到拼图图块的 URL 矩阵；接着启用多线程机制从 GoogleMap 同步取回地图图块；得到所有的图块后拼合成一个大图，然后以用户要求的中心点裁剪出用户所需图幅的地图图片。

3.3 缓存机制

由于 Google 图块并不经常更新，因而没必要每次都从 Google 取所需的图块。为了提高速度，可采用两级缓存机

制，第一级缓存将所访问过的图块保存到本地硬盘上或数据库中，并建立索引；第二级缓存则将拼合成的大图保存到本地硬盘上或数据库中，并建立索引。这样，在用户请求地图时，可先查询大图索引。若已有所需的大图，则根据用户请求地图的中心及图幅裁剪该大图即可；如没有所需大图，则查询图块索引，对已有的图块直接从本地读取，并从 Google 下载并保存没有的图块，拼合成大图并保存、裁剪即可。

基于上述算法的系统流程如图 4 所示。

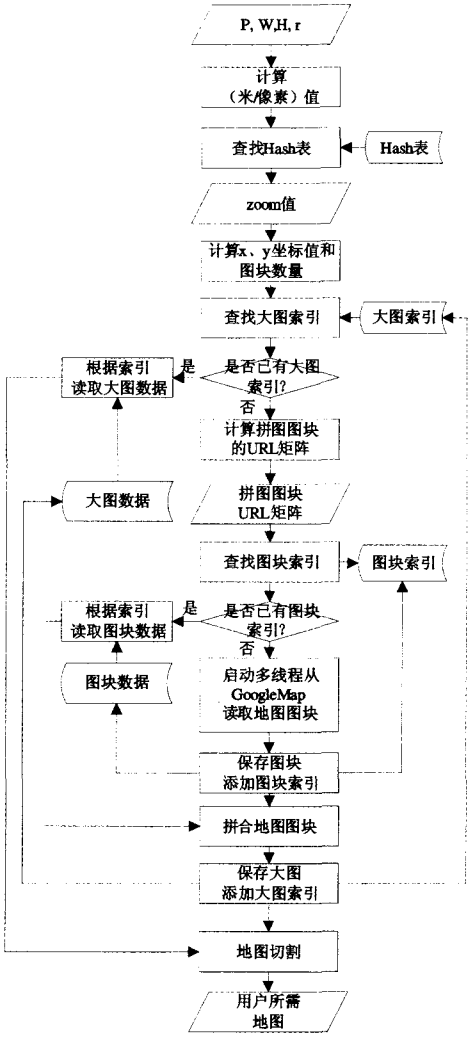


图 4　系统流程图

4　实例

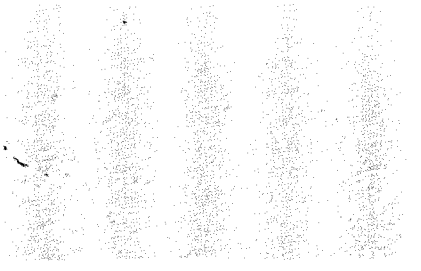
图 5 为本文所研究的 Google 地图构造系统所获得的地图实例。

其中，(1)图为北京西直门地图的卫星地图；(2)图为纽约某街区的合成地图，包含有主要街道名称标注；(3)图为赤道上且经度为极值的卫星地图，演示了经度极值地区的地图拼接效果；(4)图为南极的卫星图片，无图部分以黑色表示。

总结　本文系统地研究了 Google 地图的算法及实现，可以为我国地理信息系统（Geographic Information System，GIS）的应用集成提供借鉴。与国内的 GIS 系统相结合，它可以被广泛地应用于土地规划、环境监测、防灾减灾、地学、资源管理、电力行业、交通管理、城市规划、科研、教育和国防等领域。

(1)北京卫星图：纬度 39.941，经度 116.35，zoom=17，图幅 1024*1024

(2)纽约合成图：纬度 40.75，经度-74，zoom=17，图幅 1024*1024



(3) 赤道卫星图：纬度 0，经度 180，zoom=7，图幅 1024*1024

(4) 南极卫星图：纬度-90，经度-180，zoom=4，图幅 1024*1024

图 5　Google 地图实例

参 考 文 献

1　孙达，浦英霞. 地图投影. 南京：南京大学出版社，2005
2　Google Map API 文档. http://www.google.com
3　Bildirici I O. Numerical inverse transformation for map projections. Computers and Geosciences, 2003, 29(8):1003～1011
4　Williams R T. Lambert and Mercator map projections in geology and geophysics. Computers and Geosciences, 1995, 21(3):353～364
5　Snyder J P. Map Projections Used by the U S Geological Survey. Washington: U S Government Printing Office, 1982