

# 一种适合于地图出版符号的反走样算法研究

邓术军<sup>1</sup> 郭建星<sup>1,2</sup>

(1 信息工程大学测绘学院,郑州市陇海中路 66 号,450052)  
(2 武汉大学遥感信息工程学院,武汉市珞喻路 129 号,430079)

**摘 要:**在分析现有反走样算法的基础上,提出了一种适合于地图出版符号的反走样算法——“刷子法”,并通过实验对该方法的反走样效果和效率进行了全面评估。  
**关键词:**走样;反走样;地图出版符号;刷子法;分辨率  
**中图法分类号:**P283

在计算机图形学中,要在光栅屏幕上显示一个连续图形(如直线、折线、Bezier 曲线等),就必须将连续图形离散化,以期与构成屏幕的离散像素匹配。这种用离散量表示连续量引起的信息失真就是走样(aliasing),而用于减少或消除这种失真效果的技术就是反走样(anti-aliasing)<sup>[1,2]</sup>。

由于走样的存在,计算机屏幕上表示的图形具有两个明显的缺点:除水平、垂直和 45°角的直线外,大部分图形显示时边缘会产生锯齿;只能表示宽度整倍于像素大小的图形,对于非整倍的图形按近似的整倍情况处理。这两个问题对于追求“所见即所得”、追求符号精度的出版系统来说是不应该忽视的。但目前通用的出版系统(如 Illustrator、Freehand、CrowlDraw、Photoshop、Word 等)在显示图形时都采用了反走样技术;而国内外常用的地图出版系统和 GIS 系统(如 Microstation、MapGIS、方正智绘、ArcGIS、MapInfo 等)在显示地图时却忽视了该问题。

## 1 常用的反走样方法分析

自从 Crow<sup>[2]</sup>在计算机图形学领域中提出走样问题并给出了一种方法以来,已经有很多反走样方法问世。根据算法原理的不同,常用的反走样方法可分为以下两大类:① 提高分辨率即增加采样点,也称过取样(supersampling)、后过滤(post-filtering);② 将像素视为一个有限区域,进

行区域采样,也称为前过滤(pre-filtering)<sup>[1,3,5,6,7]</sup>。

### 1.1 提高分辨率

根据采样原理,走样是由于采样不充分而引起的信息失真,消除走样最直接的方法就是增加其采样点个数,即提高分辨率,其中包括提高物理分辨率和逻辑分辨率两种方式。

1) 提高物理分辨率就是提高显示器的分辨率。显示器的分辨率越高,显示的图形越精细,走样情况越不明显。

2) 提高逻辑分辨率是指对显示器的每个物理像素进行进一步细分,获得若干个逻辑像素,这就相当于在逻辑上提高了显示器的分辨率,在进行图形扫描转换时,确定每个逻辑像素的着色情况,然后根据每个物理像素中所有逻辑像素的着色情况,利用加权或非加权平均的方式获取对应物理像素的颜色值。图 1 描述了提高逻辑分辨率的基本原理(以直线段为例),其中图 1(a)表示其计算过程,而图 1(b)表示其显示结果,图中水平和垂直的粗实线构成的矩形表示屏幕上的实际像素,细虚线构成的矩形表示高分辨率的逻辑像素,图 1(a)中的小圆表示该逻辑像素应着色,图 1(b)中的大圆表示该物理像素的着色情况,其亮度通过计算各个物理像素中逻辑像素的非加权平均值来获取。当然,也可进行加权平均,加权平均的基本思想是不同逻辑像素的着色情况对物理像素的亮度影响程度不同,靠近物理像素中心的逻辑像

素影响力较大,边缘的影响力较小。

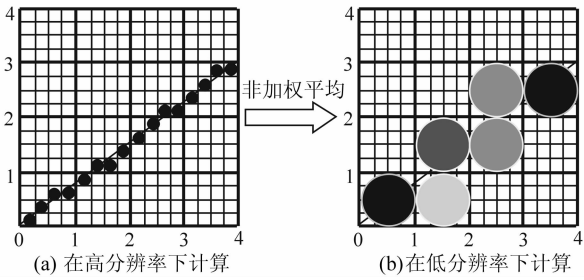


图 1 提高逻辑分辨率的基本原理  
Fig. 1 Principle of Higher Logical Resolution

1.2 区域采样

在提高逻辑分辨率的算法中,无形中假设了图形为一个数学意义上无宽度的理想线条。但事实上,线条是有宽度的,在很多情况下不能忽略,尤其在地图出版系统中,线条的宽度具有重要意义,比如同样的棕线,0.1 mm 代表首曲线,0.2 mm 就代表计曲线。在进行线条的扫描转换时,为了确定像素的着色情况,就需要计算构成像素的正方形区域与构成线条的多边形之间的空间关系。如果构成像素的正方形区域与构成线条的多边形相离,该像素不着色;如果构成像素的正方形区域被构成线条的多边形包含,该像素着线划色;如果构成像素的正方形区域与构成线条的多边形相交,该像素着过渡色,其亮度根据两者的相交部分确定,这正是这部分像素反走样产生的效果。

图 2 描述了区域采样的基本原理,每个坐标网格代表一个像素区域,中间粗线构成的多边形表示端点为(1,1)、(7,5),宽度为 1.5 个像素的直线段(折线、曲线等其他线条类似)。对于像素(0,0),因其与多边形相离,不着色;对于像素(1,1),因其被多边形包含,着线划色;对于像素(1,0),因其与多边形相交,着过渡色,所着过渡色的亮度由两者相交的部分决定(图中用晕线表示)。

对于相交像素亮度的计算可分为两种情况:① 亮度完全由相交部分面积的大小决定,亮度正比于相交面积的大小,即非加权区域采样;② 亮度不仅与相交部分面积的大小有关,还与相交部分的位置有关(相交部分越靠近像素中心,影响力越大),即加权区域采样。

2 一种适合于地图出版符号的反走样算法——“刷子法”

2.1 “刷子法”的基本原理

综上所述,提高分辨率的反走样方法非常简

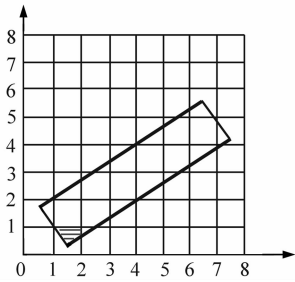


图 2 区域采样的基本原理  
Fig. 2 Principle of Region Sampling

单、易于实现,但它有以下两个缺点:① 根据提高分辨率的不同倍数,只能在一定程度上减轻走样,而反走样效果有限;② 将线条视为数学上的理想线,不能解决线条的宽度问题。与之相比,用区域采样能够获得更加理想的反走样效果,也能方便地处理线条的宽度问题,但它存在计算量大、不易实现的缺点,尤其在处理曲线时,该缺点更加明显(需要计算一个正方形区域和一个不规则面域的交集)。

作为地图出版符号,其线条必须有确切宽度,因而,直接用提高分辨率的方法进行反走样不能满足其要求。另外,地图出版符号的绘制速度与其显示质量同等重要,在保证一定显示质量的条件下应该使反走样的算法尽可能简单。由此可见,将前面介绍的两类常用的反走样算法直接应用到地图出版符号中都不太合适。笔者结合前面两类算法的思想,提出了一种适合于地图出版符号的反走样算法,不妨称作“刷子法”(相当于用一个圆形刷子沿中心线刷一遍)。

图 3 描述了“刷子法”的基本原理,该图表示的是一条任意曲线  $f(x,y)$ ,线宽  $w$  为 1.5 个像素,线的首末点分别为(1,1)和(5,5)。利用“刷子法”实现  $f(x,y)$  的反走样处理需要如下 3 个基本步骤:① 提高显示器的逻辑分辨率,在高逻辑分辨率条件下对图形的理想中心线进行扫描转换,获得描述理想中心线的离散逻辑像素,在图 3(a)中用小的实心点表示。② 沿着理想中心线上的离散逻辑像素,移动一把圆形“刷子”,圆心与各个离散逻辑像素重合,直径为线宽度,将落入任何一个圆形“刷子”中的逻辑像素标记为“选中”,图 3(a)中用小的空心点表示(图中只标出了物理像素(0,0)和(5,5)中被“选中”的逻辑像素)。③ 根据每个物理像素中逻辑像素被“选中”的情况,计算该物理像素的着色情况。如果没有逻辑像素被“选中”,该物理像素着背景色;如果所有逻辑像素都被“选中”,该物理像素着线划色;如果部

分逻辑像素被“选中”，该物理像素着过渡色，计算结果如图 3(b)所示。

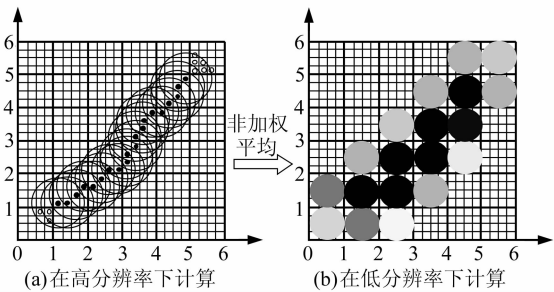


图 3 “刷子法”的基本原理  
Fig. 3 Principle of Brush Method

2.2 “刷子法”算法的具体实现

1) 在高分辨率下获取描述理想中心线的离散逻辑像素。首先指定逻辑分辨率提高的倍数  $R$ 。 $R$  越大, 过渡色的等级越多, 反走样的效果越好, 但  $R$  每提高一倍, 帧缓存容量增加到原来的 4 倍, 扫描转换的时间增加将近 4 倍。由此可见, 反走样的效果和效率是矛盾的, 在确定  $R$  时, 应注意反走样效果和效率的平衡。就地图出版符号而言, 将  $R$  指定为 4 可以获得比较理想的平衡效果, 本文在实验中采用的  $R$  值为 4。

确定  $R$  值后, 将基于屏幕物理分辨率的物理坐标系变换成基于逻辑分辨率的逻辑坐标系。与之对应, 物理坐标系下描述线划的函数  $f(x, y)$  变换成逻辑坐标系下描述线划的函数  $F(X, Y)$ , 其中  $X=R \times x, Y=R \times y$ 。根据  $F(X, Y)$  所代表线划的具体类型, 采用不同的转换方法(如直线采用 Bresenham 画线算法, 圆采用中点圆算法, 椭圆采用中点椭圆算法等), 获取描述  $F(X, Y)$  的离散逻辑像素(图 3(a)中小的实心点)。其中, 各种线划的扫描转换方法是计算机图形学中的一个基本问题, 可参考文献[1, 5]。

2) 逻辑像素“选中”与否的判断。在逻辑坐标系下, 假设描述  $F(X, Y)$  中心线的离散逻辑像素为  $(X_i, Y_i) (i = 1, \dots, n, n$  为中心线上离散逻辑像素的个数, 图 3 中  $n$  为 20)。以  $(X_i, Y_i)$  为圆心、 $W$  为直径( $W$  为逻辑坐标系下线划的宽度,  $W = R \times w$ , 图 3 中  $W$  为 6), 构建  $n$  个圆形“刷子”, 将落入任何一个“刷子”中的逻辑像素标记为“选中”(图 3(a)中小的空心点)。

3) 物理像素着色情况的计算。在物理坐标系下, 统计出每个物理像素被“选中”逻辑像素的个数, 根据统计结果计算各物理像素最终的着色情况。假设物理像素  $(x_j, y_k)$  中被“选中”的逻辑像素有  $K (0 \leq K \leq R^2)$  个, 则物理像素  $(x_j, y_k)$  最

终的着色为:  
结果色 = 背景色 +  $K/R^2 \times (\text{线划色} - \text{背景色})$ ,  
 $K \in [0, R^2]$  (1)

为了计算机实现的方便, 将颜色用  $R、G、B$  三刺激值表示<sup>[4]</sup>, 式(1)可以转换为:

$$\begin{cases} R_{\text{结果色}} = R_{\text{背景色}} + K/R^2 (R_{\text{线划色}} - R_{\text{背景色}}) \\ G_{\text{结果色}} = R_{\text{背景色}} + K/R^2 (G_{\text{线划色}} - G_{\text{背景色}}) \\ B_{\text{结果色}} = R_{\text{背景色}} + K/R^2 (B_{\text{线划色}} - B_{\text{背景色}}) \end{cases} \quad (2)$$

式中,  $K \in [0, R^2]$ 。

3 实验分析

实验的基本平台为普通 PC 机(PⅣ, 内存为 512 MB, 显卡 GeForce4, 显存 64 MB), 编程环境为 VC6.0。实验的目的是检测采用“刷子法”构建具有反走样效果的地图出版符号的效果和效率。

实验 1 测试不同宽度线划的反走样效果。图 4 是对用于制作地图出版符号的线划表采用“刷子法”进行反走样处理的结果, 以 mm 为单位, 是参照总参测绘局发布的点线符号标准表制作的。从图中可以看出, 由于采用了“刷子法”进行反走样处理, 使得描述线划的像素不只具有着色和不着色两种情况, 还有着过渡色的情况。正是过渡色的存在, 克服了原来只能表示宽度整倍于像素的线划这一缺点, 可以用过渡色的浓淡变化来描述任意宽度的线划。

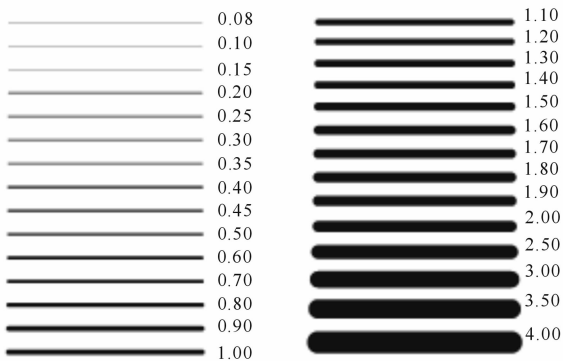


图 4 点线符号标准表(等大)  
Fig. 4 Criterion of Point and Line Symbol (Isometry)

实验 2 测试地图出版符号的反走样效果。图 5 是部分地图出版符号在反走样前后的对比, 为了便于比较, 将符号进行了图形放大, 使得 0.1 mm 宽的基本线划刚好用一个物理像素点表示。从图中可看出, 反走样后的符号比较精细, 基本看不到锯齿现象。

实验 3 测试“刷子法”的效率。从屏幕左上

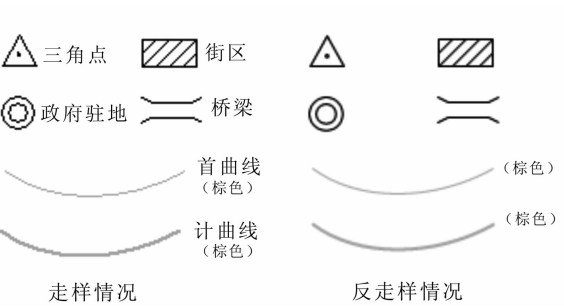


图 5 部分符号反走样前后对比(放大)  
Fig. 5 Comparison of Some Symbols Before and After Anti-aliasing(Amplification)

角向右下角重复绘制 1 000 条直线,表 1 记录了反走样前后所需的绘制时间,表中不同列代表不同的线宽(以像素为单位),测试的最大线宽为 11(相当于 4 mm,也就是图 4 中表示的最大线宽),时间单位是 ms。从表 1 可看出,反走样前,宽度为 1 的基本线划绘制速度非常快,其他宽度的线划绘制速度基本相同,随宽度增加耗时增加非常少;反走样后,宽度为 1 的基本线划绘制速度比反走样前慢 13.8 倍,其他宽度的线划绘制耗时随宽度匀速增加。对于地图符号用到的线宽,反走样后与反走样前的耗时比在 5.4~17.8 之间。从表 1 还可以间接地得出,在屏幕上显示一幅清晰的地图,采用“刷子法”进行反走样处理后,其绘制时间一般不会超过 3 s(心理学上认为,3 s 为人们等待屏幕刷新的时间极限)。

表 1 “刷子法”的效率

| Tab. 1 | Efficiency of Brush Method |       |       |       |       |       |
|--------|----------------------------|-------|-------|-------|-------|-------|
|        | 1                          | 3     | 5     | 7     | 9     | 11    |
| 反走样前   | 63                         | 234   | 235   | 250   | 250   | 250   |
| 反走样后   | 875                        | 1 625 | 2 282 | 2 953 | 3 688 | 4 453 |
| 耗时倍数   | 13.8                       | 6.9   | 9.8   | 11.8  | 14.8  | 17.8  |

参 考 文 献

1 孙家广. 计算机图形学. 北京:清华大学出版社,1998  
2 Crow F C. The Aliasing Problem in Computer-generated Shaded Images. Communications of the ACM, 1977,20(11):799~805  
3 Fabris A E, Forrest A R. Antialiasing of Curves by Discrete Pre-filtering. SIGGRAPH '97, Campos do Jordao, Brazil, 1997  
4 胡成发. 印刷色彩与色度学. 北京:印刷工业出版社,1993  
5 Hearn d, Baker M P. 计算机图形学. 蔡士杰,吴春镭,孙正兴,等译. 北京:电子工业出版社,2002  
6 张波,张焕春,经亚枝. 罗盘刻度线反走样快速绘制算法的改进研究. 计算机辅助设计与图形学学报, 2003(1):71~75  
7 Fabris A E, Forrest A R. High Quality Rendering of Two-dimensional Continuous Curves. SIBGRAPI '97, Campos do Jordao, Brazil, 1997

第一作者简介:邓术军,博士生,主要从事空间数据模型与空间数据引擎方面的研究。  
E-mail:chxydsj@163.com

An Anti-aliasing Algorithm Suitable to Map Publishing Symbol

DENG Shujun<sup>1</sup> GUO Jianxing<sup>1,2</sup>

(1 Institute of Surveying and Mapping, Information Engineering University, 66 Middle Longhai Road, Zhengzhou 450052 ,China)  
(2 School of Remote Sensing and Information Engineering, Wuhan University, 129 Luoyu Road, Wuhan 430079,China)

**Abstract:** On the basis of analysis of various algorithms, an anti-aliasing algorithm called brush method is presented in this paper, which is suitable to map publishing symbol. After introducing the basic principle and implementation of Brush Method in detail, its effect and efficiency are evaluated through experiment.  
**Key words:** aliasing; anti-aliasing; map publishing symbol; brush method; resolution

About the first author: DENG Shujun, Ph.D candidate. He majors in spatial data model and spatial data engine.  
E-mail: chxydsj@163.com