

111

国际图形会议译文集

B类 1 册

武汉工业控制计算机外部设备研究所
情 报 室

目 录

光栅式CRT显示的图象毛刺消除	1
数字逻辑模拟系统的交互图形处理器	12
用B-仿样面和CAD/CAM系统进行船壳设计	30
三维色调图形的交互技术	43

光栅式CRT显示的图象毛刺消除

日本东京Graphica计算机公司

Wieslaw Romanowski

摘要

本文就光栅式CRT显示的线条画面和连续色调图象合成过程出现的毛刺现象,提出一种消除它们的算法。这种算法产生的图象不仅在质量上与所用最完善技术所产生的图象媲美,而且算法的设备和计算的成本极低,通过大量的实验结果和成功的实际操作证明其算法具备这些特点。

在实现算法的过程中,可消除这两种毛刺现象,其一用匀滑矢量产生器可产生匀滑的线条画面;其二用消除毛刺后置处理器可解决连续色调图象输出的毛刺问题,这两者都是显示系统具有智能的特征。

由于该算法的通用性,速度、精度和廉价的设备,使此算法能廉价地固化,这样在实用中,成为消除毛刺的一种有效方式。

1、前言

由于近年来存贮器成本急剧下降,导致光栅扫描显示器的发展,使它与随机扫描显示的竞争中能遥遥领先。光栅扫描这些特征有利于图象的真实感,这样把图象应用推进了新的领域。另外,它和已经研制成功的随机扫描一样,可用于很多方面。然而就其应用而言,不可避免地碰到众所周知有关器件特征的问题:毛刺和光栅问题。人们往往把由于毛刺引起图象质量下降认为是光栅显示装置的唯一弱点,所以在克服显示的线和面出现的“毛刺”或梯状时,花费了相当的精力。这样所有解决问题的方案都集中在显示器的帧缓冲寄存器上。因为帧缓冲寄存器可控制象素的强度(可控制高于2个强度/每个象素)。任何算法,无论它是采用哪一种途径,最后归结到每个象素的灰度等级。已经提出过许多复杂程度不同的近似方法,然后在现行采用的方案中,要在可能情况下产生高质量的消除毛刺的图象,但是由于缺乏通用性或涉及到大量的计算问题,所以从实用的观点来看,耗资太大,速度太慢。任何一种消除毛刺技术成功的关键,除质量外,再是速度问题。本文所介绍的算法,就在产生高质量消除毛刺的同时,成功地解决了速度问题。

2、空间滤波

就光栅式CRT显示的“毛刺线”而言,人们很清楚地意识到不存在用药物试剂解决毛刺问题而没有付作用。这点可以直接用所谓的“取样理论”说明,即被取样的信号,其频率高于取样频率半倍以上,就不可能使这种频率向量再现。

根据这一理论解决毛刺问题,在取样之前采用不同的滤波技术。这种滤波目的是滤掉图象的空间频率。因为这种频率太高,不能在已定的光栅上显现出来。很明显,空间滤波消除毛刺是以损失空间频率为代价的。因滤波中滤掉的高频损失在屏幕上显示的图象或多或少有

点模糊。换句话说,在取样前用低频滤波可改变图象的质量。在处理光栅式CRT显示的矢量时,认识这点是特别重要的。在不采取任何滤波的情况下,对相对宽度的矢量保持高的空间频率没有消除图象毛刺重要。滤波程序涉及到选择数学方式控制滤波器的函数。假设矢量宽度是一定的,那么只要把滤波器函数对所复盖的数域进行积分,就可得到实际需要矢量的每个像素的强度。

另外,Nishida和Nakamae提出的一种算法是使像素强度与构成矢量的数域成比例。这指的是用傅立叶窗口作为滤波器函数,此函数以各种形式交叉线,这样计算结果都包含在傅立叶窗口的有效范围的计算数域内,实际上这种算法是不理想的。而S. Gupta和R. F. Sproul采用一种圆锥形的滤波函数。因为它对给定宽度的线具有环形对称,仅只依据像素中心到中心线的距离就可算出像素强度,也就是说只要有定距离数值就可算出。这就要求对每种灰度建立一个对照表,并有说明,此表必须参照各个像素产生的情况制成。此外这种算法要求对矢量的端点进行单独的地耗时处理,加之交叉线是左开口也是个问题,更重要的一个方面是这种算法还没单独地做出实验结果证明其实用性。

3、算法

本文所提出的算法方案是利用矢量的特有的性能,即它的斜率以及在几何图形上等于零的宽度。把傅立叶窗口作为空间滤波器,假设傅立叶窗口的像素数域是 ± 0.5 ,则其数值是1.0,而其他地方均为零。在此执行的操作中,将滤波器函数正规化,那么围封的面积为1(看图1)。另外假设窗口的规格(d)就是矢量斜率的函数,那么其取向要调到如图1位置的矢量斜率,每个像素的强度等于滤波函数与线强度的卷积。滤波器函数的积分将在 $2(d-dx)d$ “区域内求得(看图1),方法极简单。每个像素强度与矢量成反比例,譬如 $C' = C(d-dx)d$,此处C是总矢量强度,C'是求价中的像素强度。

图2所示为像素强度在与矢量中心线平行的方向上的分布,当像素与矢量中心线的距离愈近,其强度愈和总矢量强度辐合。假设矢量宽度一定,而滤波器函数大小等于或小于矢量宽度,那么得出的结果是满意的。

将在oxy座标的第一象限全面深入讨论矢量问题,这样其它数域涉及值就不成问题了。首先要产生宽度尽可能小的无毛刺矢量,这样显示矢量的宽度由尽可能小的傅立叶窗口所限制(看图2)。如上所述,各个像素强度是矢量中心线距离的线性函数,根据其叠加位原理,不需要对此距离进行直接计算,用递增法求得各个像素强度。在此算法中采用像素强度的递增算法,如图4所示。标一,标二表示像素对应于矢量中心线在左边与右边的变化关系,那就是说一对像素,无论在什么情况下均在与矢量最大位移垂直方向的一行内。第一像素的x轴代表的就是这个方向。

输入数据以及矢量强度C为矢量初始端(XS1,YS)和终端(XE,YE)的座标。这个算法是由两部分组成,初始部份和主要部份,初始计算程序如下,把矢量的初始端座标取整,即可求出右像素(IX1,IY1)的座标,然后检测第一像素(IY1)Y座标上的矢量斜率(Q)和数值(A)。Y座标最后校准后,就可根据一个像素在它的最大位移方向上的增量(CP)算出强度增量。根据对应左像素初始强度计算的数值(C2)就可完成初始化的程序。每次增量只产生两个像素的矢量,我们用 $C = C_1 + C_2$ 立即算出右像素强度。左像素的座标系数仅在Y座标上有差异: $IY_2 = IY_1 + 1$,这样就把一对象素的座标系数和强度送到帧缓冲寄存器。然后以下列方式进行算法的主要部份。通过上述的(CP)就可对左像素(C2)

的强度增量，并对照它就可得出最大可能值〔如总矢量强度(C)〕。如果这个实际值仍小于(C)，那么只要在最大的运动方向调整，否则就需要在垂直的方向增加增量，然后减去总矢量强度，就得到准确的左象素强度。在算法主要部份执行的象素座标产生程序时类似Bresenham的算法。然而本文所介绍的算法是象素强度控制的变化C2直接支配它的座标产生。这意味着这种变化同时包含两个不同的含义，即物理变化和几何形的变化。而在Bresenham算法里，控制所测量垂直于最大位移轴的误差只具有几何意义。

至于谈到扩大的傅立叶窗口尺寸的算法演变如图4b所示，此种情况每行画影线只3个象素，而外部的象素强度在总矢量强度的0和2/3之间变化。而处于中间位置的象素强度变化是在总矢量强度的2/3和它的最大值之间波动。

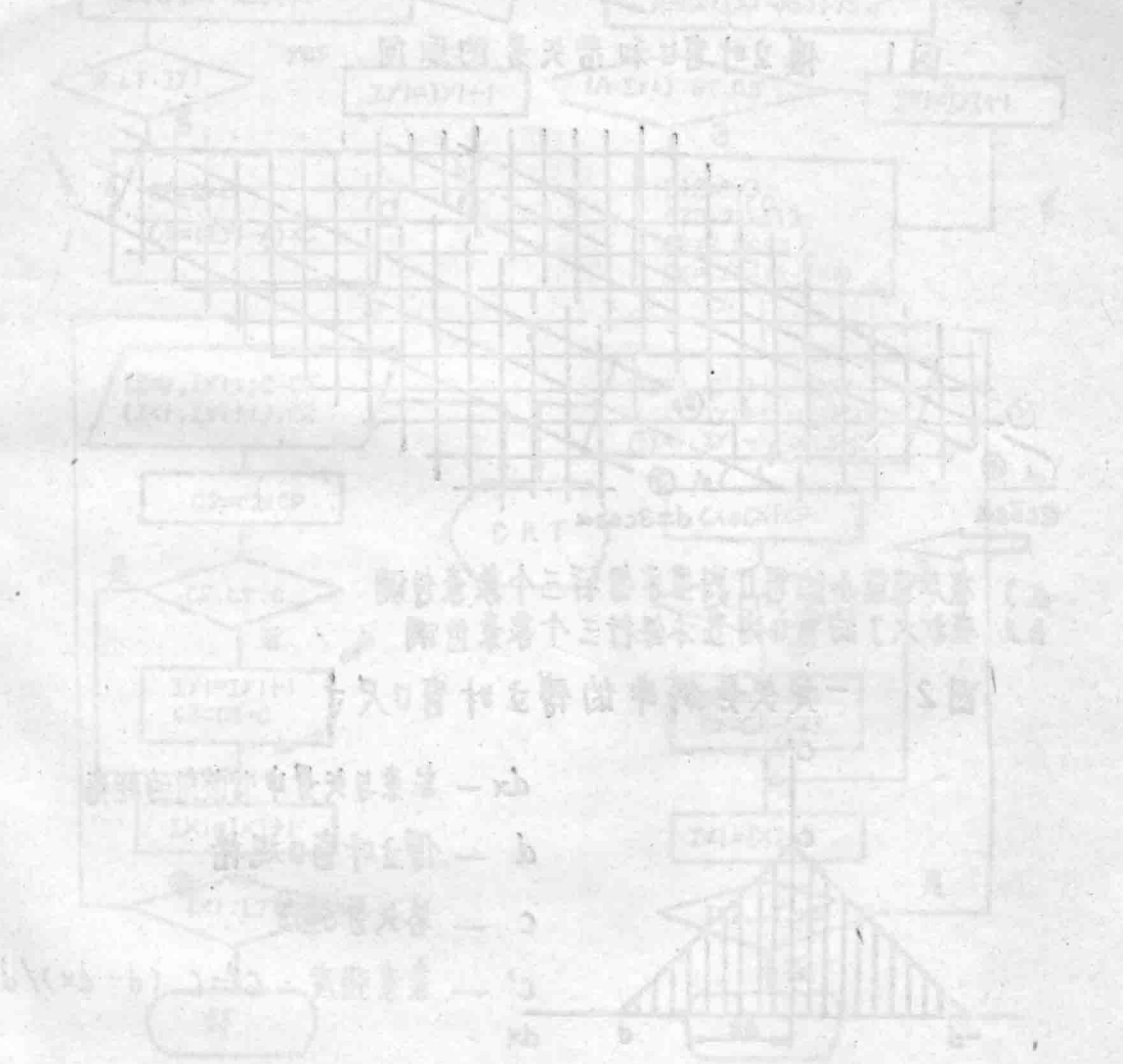


图4 算法流程图

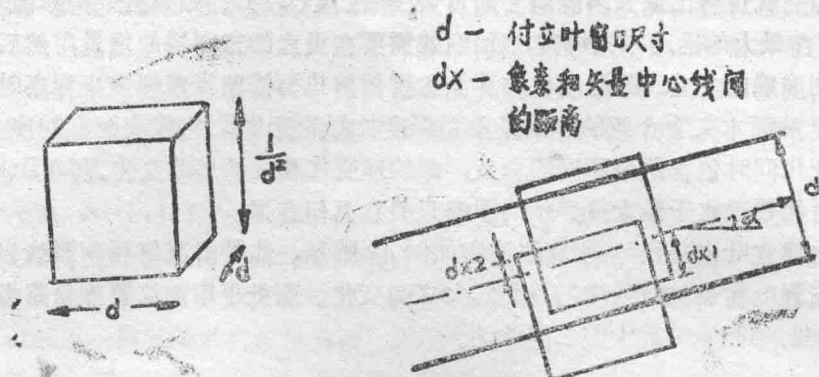
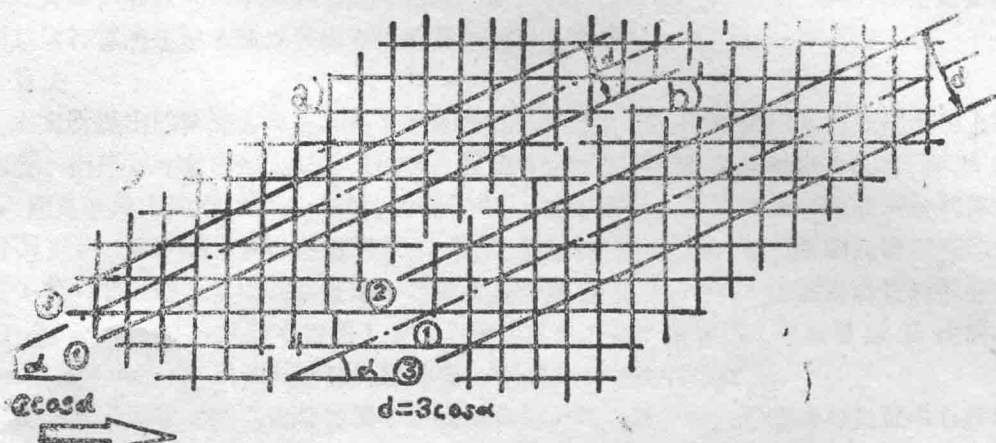
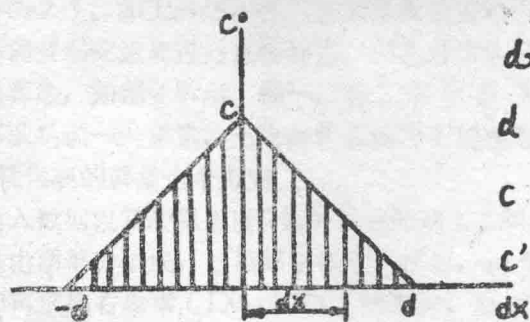


图1 傅立叶窗口和沿矢量的取向



- a) 在尽可能小的窗口内显示每行三个象素色调
 b) 在扩大了窗口内显示每行三个象素色调

图2 一定矢量斜率的傅立叶窗口尺寸



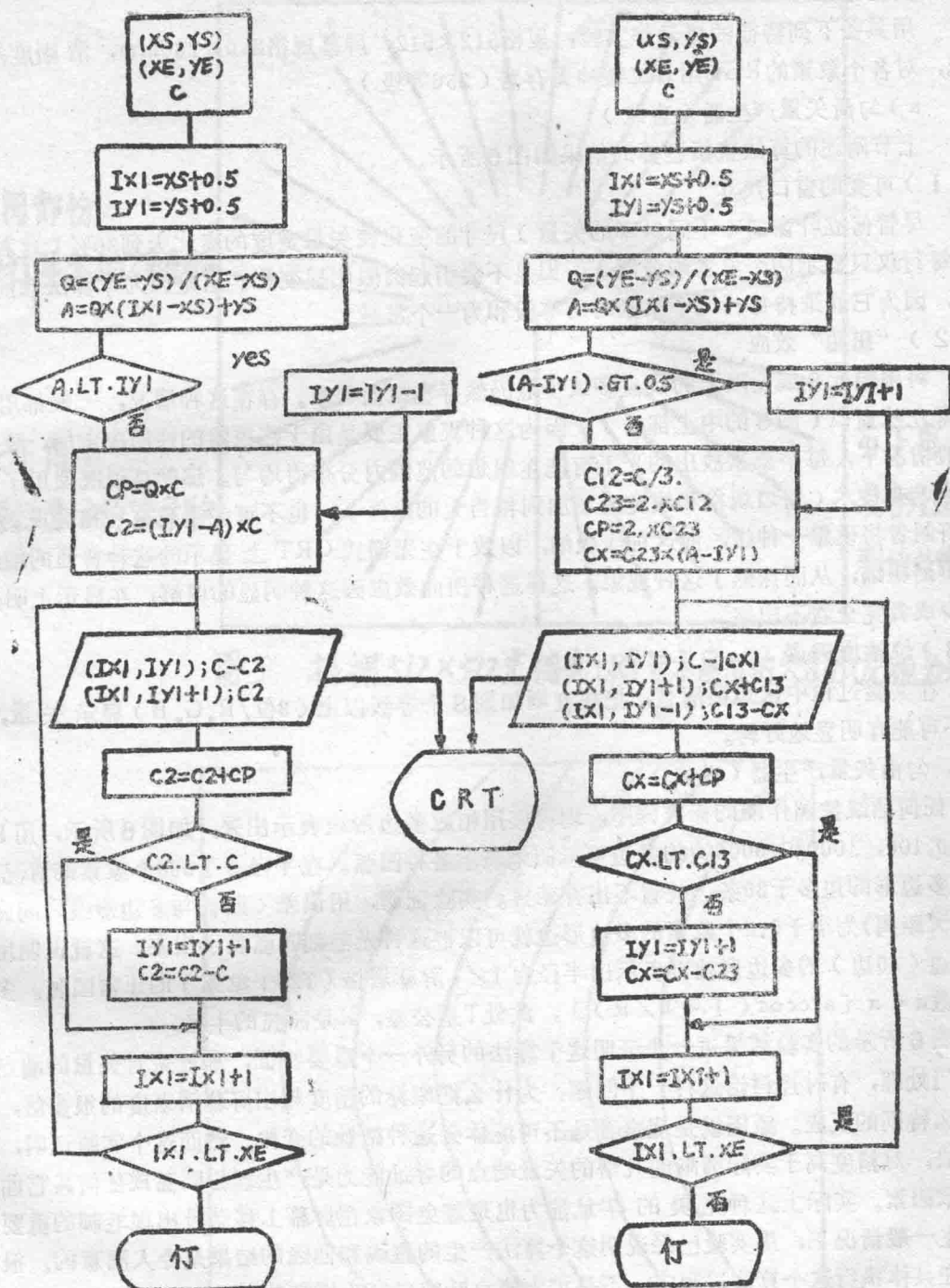
dx — 象素与矢量中心线间的距离

d — 傅立叶窗口规格

c — 总矢量强度

c' — 象素强度: $c' = c (d - dx) / d$

图3 通过矢量的象素强度分布



a) 尽可能小的傅立叶窗口；显示 2 像素/每行

b) 扩大的傅立叶窗口；显示 3 像素/每行

图 4 补法流程图

4、实验结果

用具备下列特征的显示作实验：规格 512×512 ，屏幕规格 $280 \times 280\text{mm}$ ，清晰度范围稍低，对各个像素的RGB用8位缓冲寄存器（256等级）。

a) 匀滑矢量产生器（直线）

上节阐述的算法直接运算的结果如图5所示。

(1) 可变的窗口尺寸

尽管傅立叶窗口（不同斜率的矢量）尺寸的变化使矢量宽度的变化大到30%（这种情况是每行仅只显示两个像素的色调），但是不会引起肉眼明显变化。这是因为本算法性能决定的，因为它能维持每行两个像素对的等级和为一个常数。

(2) “扭曲”效应

斜率趋近0或无限大的线或多或少地仍然存在扭曲现象。存在这种情况，一般都用较大的傅立叶窗口（图5的中上部）。因为这种现象主要是由于监控器的性质决定的，譬如在一般情况下，每个像素放出的光不可能在理想的范围内分布得均匀。那就是说既使用了任何消除毛刺技术（窗口规格和线宽度增加到相当大的除外），也不可能避免这种扭曲现象。但是研制者将设想一种线，而这种线极细，以致于在光栅式CRT上显示的这种普通的细线比较不出粗细，从而抹煞了这种现象。这样这种扭曲效应因这种明显的调解，在显示上明显地减少或者完全看不出来。

(3) 位浓度因素

在实验过程中这点很清楚，把灰度增加到8个等级以上（3位/R.G.B）显示矢量，情况不可能有明显地好转。

b)，匀滑矢量产生器（曲线）

任何曲线按照作图的标准程序，均可以用相应多边形线表示出来，如图6所示，用8，20,30,100,1000和10000边的多边形近似地表示各种圆弧。在半径等于320个像素的情况下，只要多边形的边多于30条，就看不出有差异。实验证明，用误差（圆弧与多边形线之间允许的最大距离）为小于0.2个像素的多边形边就可以把这种无毛刺圆弧表示出来。这就说明用相对少边（40边）的多边形就可表示出半径为 $1/4$ 屏幕规格（125个像素）的正常圆弧。多边形边数 $n = \pi [\arccos(1 - T/R)]$ ，此处T是公差，R是圆弧的半径。

图6所示的实验结果进一步证明这个算法的另外一个重要特征，那就是对矢量的端点无需专门处理，有时还讨论这样一个问题，为什么把座标的精度超出屏幕清晰度的很多倍，也没什么特别的反应。原因就是观察者是不可能体会这种微妙的变化。然而这个实验证明，这种算法，对精度高于实际清晰度几倍的矢量端点的寻址能力是产生理想圆弧或任何其它曲线的根本因素。实际上这种完美的寻址能力也是避免图象在屏幕上移动时出现毛刺的重要因素。在一般情况下，用实验已经说明这个算法产生的直线和曲线的结果是令人满意的。很清楚，在具体执行这个算法过程中，不是扩大傅立叶窗口产生模糊的宽线，就是选择可能最小窗口（每行显示两个像素的色调）。这种算法是在选择高空间频率和消除毛刺之间折衷的最佳方案。值得注意的是这种选择所用的线比标准矢量产生器所产生的线细得多。

中间部份:

每行显示的2象素

左下角:

每行显示的2象素

右下角

没有经过消除毛刺的处理现象

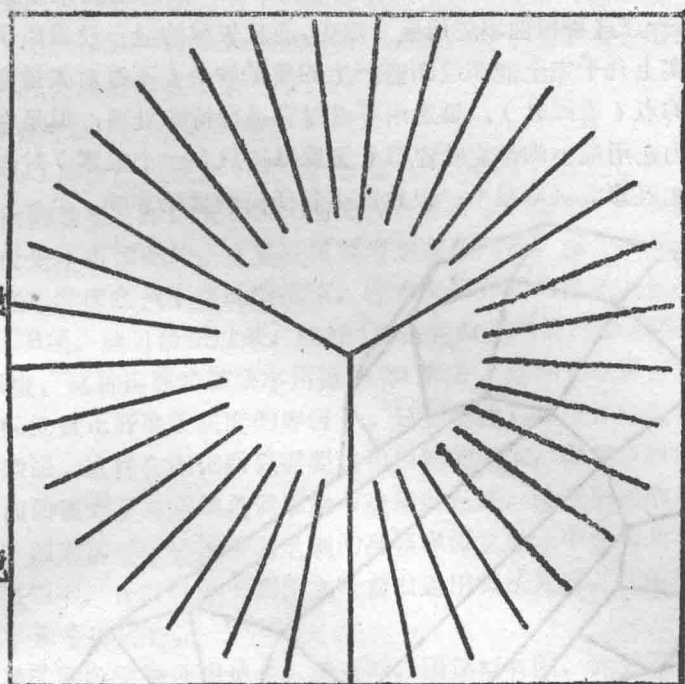


图5 根据512×512光栅式CRT显示画出的矢量图(原版为彩图)

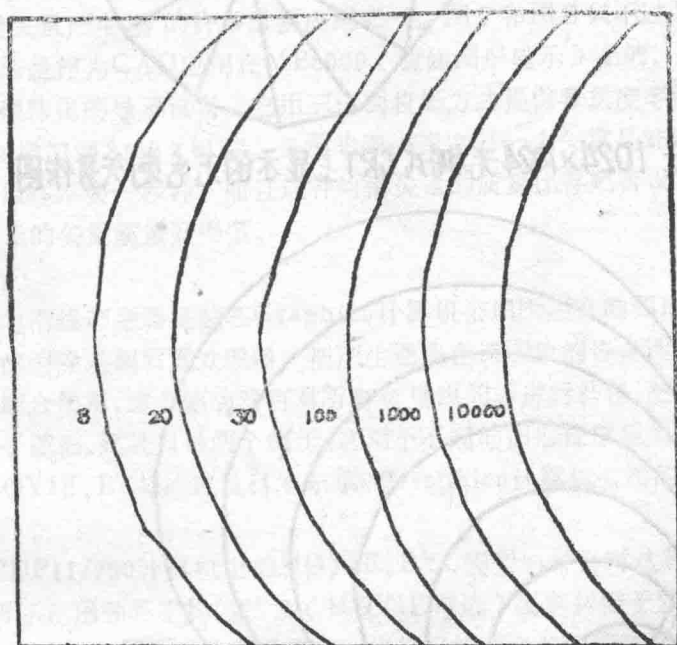


图6 在512×512光栅式CRT显示上画出的无毛刺矢量:产生曲线的扩展图(原版为彩图)

下一节将讨论实际的执行操作，然而分辨率为 1024×1024 的显示结果也应在此讨论（看图7和图8）。第一点这种扭曲率应完全消失，而且矢量的显示质量似乎比存贮管显示的一般矢量要好，事实上几乎完全能和绘图器产生的质量媲美（不过对矢量宽度除外）。另外也应注意矢量交叉的点（看图8），虽然用不着对它进行特殊处理，但是在实际操作中，也不能粗心大意。因为在使用最小的傅立叶窗口（矢量每边只有一个象素）时忽略不计是正确的，在交叉点周围的相互影响区域很小，以致没引起任何明显的影响。

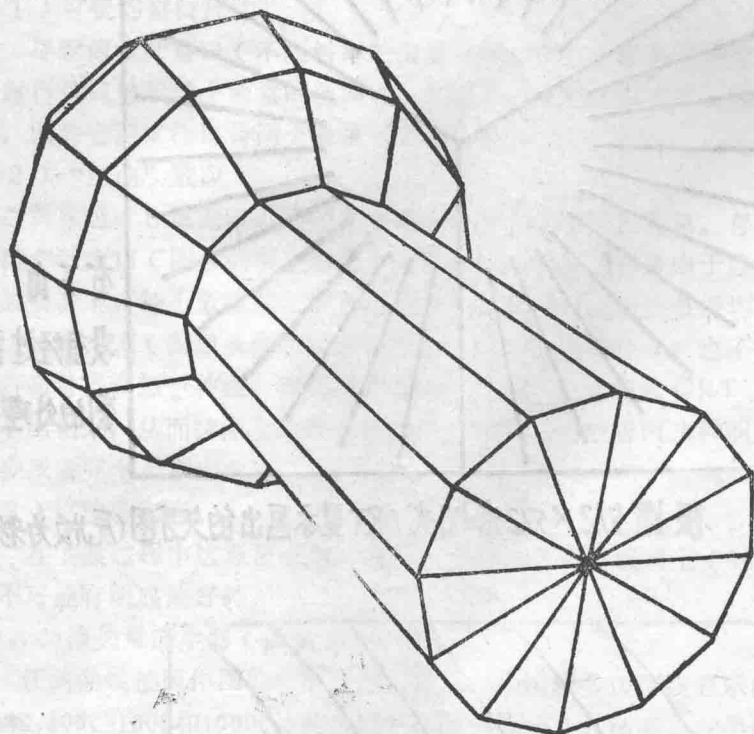


图7 在 1024×1024 光栅式 CRT 上显示的无毛刺矢量作图

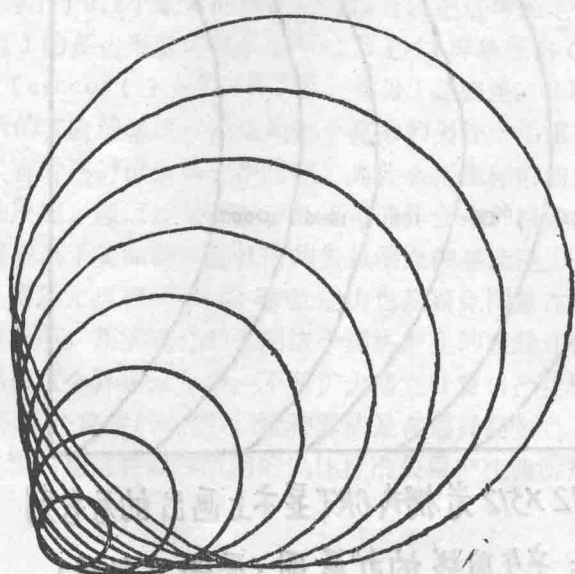


图8 在 1024×1024 光栅式 CRT 上显示的无毛刺矢量作图

据本文作者所知,在日本和美国对于消除毛刺的矢量产生器的若干实验结果和实际过程,实际上的效果不如Nishida和Nakamae所提的方案,除了矢量由于用目前算法明显窄以外,其实验与Nishida方法的效果不相上下。

c), 消除毛刺后置处理器(匀滑线产生器)

前面(a)和(b)部份已经谈到了在考虑底色的情况下,不会产生令人满意的效果问题。此处“底色”指的是在实际作矢量图之前出现在屏幕上的任何一种影象。就算法而论,底色不均匀,其情况是不可预测的。在算法里要考虑底色问题,涉及到所有一系列象素产生的强度数值读出。在非考虑底色不可的情况下,在本文所介绍的算法中要在各个象素强度加进 $(1 - CI/C) \times CB$ 项,就可修正过来,此处CI是在改正之前的象素强度,C是总的矢量强度,CB是底色强度。这种运算步骤要求用除法和乘法,这两种运算步骤计算的精度较低(要考虑进去的实际位数正好等于灰度的等级)。只要稍微对算法作些改动,就可把这种除法和乘法换成两种乘法。这样在固化后只需要用相当低的成本,把底色问题考虑进去了的效果如图5所示。在前面谈到了对连续色调图象多边形的色调,涉及到边的匀滑,看来关键还是底色考虑的问题。因为这对可能保持无毛刺的高频率图象显示中是极为重要的,在进行匀滑矢量的执行操作过程中,在尽可能小的傅立叶窗口运用算法处理,任何显示的一行都要把两个邻接多边形的象素考虑进去。

用匀滑线产生器的实验结果证明是令人满意的,因篇幅有限,此处不再引叙,下一节再介绍其实际应用。

5、应用

a) 匀滑矢量产生器

CAD是匀滑矢量产生器的许多重要应用之一。图7和图8只不过是其中一个例子。第三节里介绍的算法是作为CAD应用在MB8000(智能图形显示)上的。MB8000是一种分辨率为 1024×1024 硬件化的显示设备。它用三位面投影方式提供8灰度等级。这样寻址的点数目从 1024×1024 增加到 8192×8192 。矢量的画线速度对一个矢量是256象素为 $732\mu s$,则画出1000个匀滑矢量还不要一秒钟。而且这种匀滑矢量的质量比存贮管设备(所用的清晰度为 4096×4096)显示的矢量质量好得多。

b) 匀滑线产生器

上面所讲的匀滑线产生器是装在Graphica计算机公司所生产的图形显示M508里,执行操作作为一种智能消除毛刺后置处理器。把产生连续色调图象的许多图形组件通过驱动器与这个后置处理器配合使用,这个驱动器可对所有可见线的表进行补偿,然后把它传输到匀滑线产生器里。限于篇幅,此处只举两个例子。这两个不同的图形程序是美国犹他州的Brigham Young大学的MOVIE.BYU,以及日本东京的Graphica计算机公司的Luminous(分别于图9和图10所示)。

装在VAX PDP11/750计算机上的MOVIE.BYU需要31秒钟对这种有毛刺的图形进行计算,如图9a所示,用预算平均值技术(MOVIE可选)要求两倍于显示的分辨率进行计算。当然消除毛刺不是能充分地满足要求的,仅仅只能减少毛刺的现象。然而这种局部范围内消除毛刺,每帧需用55秒,它用的时间约为图形计算时间的170%。这样就要在连续色调图形计算过程中找个位置贮存可视边的表。然而这种匀滑线产生器实质上不要一秒钟,就可对图9a所示的图进行后置的消除毛刺处理,其结果如图9b。用装在VAX 11/750计算机上

的Luminous对图10a所示的图形进行计算，需要一分半钟。利用其消除毛刺功能，要求三倍于显示的分辨率进行计算，其显示结果虽然满意，时间却用了4分钟。而只要考虑用这种匀滑矢量产生器几乎不花时间，就可得到如图10b一样的结果。

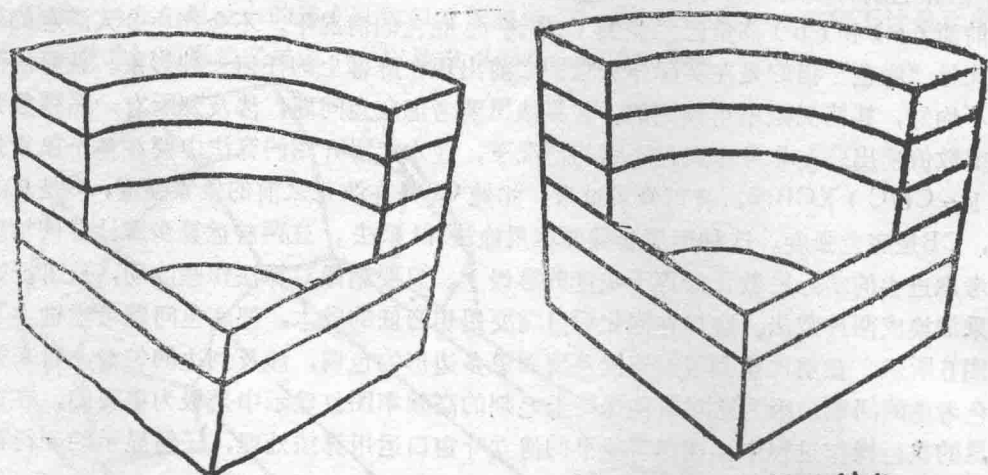
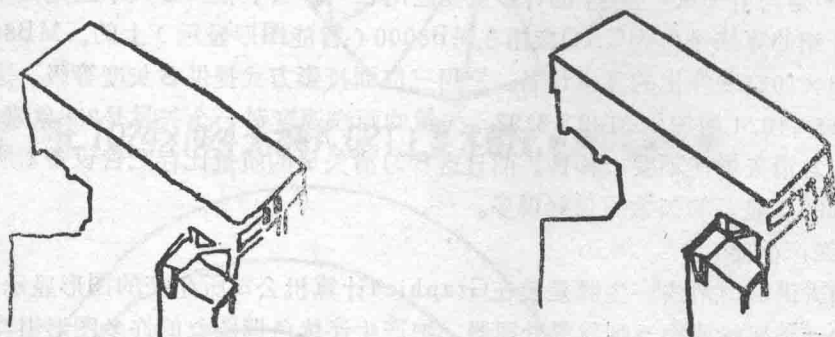


图9 通过 512×512 光栅式CRT显示的MOVIE.BYU输出的连续色调图象(原版为彩图)

a) 在消除毛刺以前

b) 在当地完成的后置毛刺处理后的显示



通过 512×512 光栅式CRT显示的Luminous的输出连续色调图象(原版为彩图)

a) 在消除毛刺之前

b) 在后置消除毛刺后

有暗影的边未经处理,从说明在显示上作为当地处理的消除毛刺情况,证明其执行操作的智能特征

6、结束语

本文就光栅式CRT显示中消除毛刺算法的研究和实际执行作了说明。这种算法在实际操作中可提供匀滑矢量产生器和匀滑线产生器。在研制消除毛刺产生器中涉及很多问题：譬如矢量端点的寻址能力，底色的处理，矢量的交叉和矢量质量的讨论，并以实验结果说明它是一个切实可行的方案，多次应用性能优良，这个算法能产生高质量的消除毛刺的图象，其处理速度高，而且执行操作和计算成本低，还有通用性。这些理想的特征是使此算法广泛应用的依据。

参考文献：

- (1). William M. Newman, Robert F. Sproul Principles of Interactive Computer Graphics MCGraw-Hill Kogakusha, Ltd.
- (2). T. Nishida, E. Nakamae
Jag Removal Method for Lines on Color Displays
(Japanese)
Computer Vision 10—1(1981.1.22)
- (3). Satish Gupta, Robert F. Sproul
Filtering Edges for Grag—Scale Displays Computer Graphics Vol.15, No3
August 1981
- (4). Blinn, James Frederick
Computer Display of Curved Surfaces University of Utah, PH. D, 1978

余碧云译

李志德校

数字逻辑模拟系统 的交互图形处理器

国立台湾大学

余慧吉 冬才林

摘要

计算机图形的发展甚至与计算机的发展相比仍是引人注目的。许多计算机辅助设计系统已把图示作为一种重要工具。计算机图示可向设计者提供一种可见图形输入输出方式,这种方式比常规的方式使用户更方便和易于理解。本文就数字逻辑模拟系统的灵活交互图形处理器的设计加以介绍。要使设计的输入方式不仅简单,用户方便,还应具备纠错和菜单驱动的方式。而提供这种功能很强的图形编辑能力,不仅能使用户作图,擦除,复制,而且能移动示意图的符号,标志,线以及信息块,而使用的图形处理器的有关数据库支援不限于一个模拟系统。

前言

工程师对计算机分析大量工程问题的卓越能力有所了解,而且开发了很多程序用到工程方面。但是不是所有的程序都是工程师普遍能解决的。在计算机辅助设计系统中不可否认的主要问题或许是由于设计工程人员对输入数据定义了解得不充分,因此不能对这种冗长的易错的数据作好准备造成的。

一般常规的模拟系统总是接收设计的数据卡片,而作图设计就其性质来说是一种图示过程。按照额定的程序,设计者必须把设计从图示方案数换成数据字母串,然而在转换过程中,错误总是难免的,因为字母串不可能像设计图案,使人一目了然,所以数据卡片的错误也不能像设计图案一看就觉察出来。

提供一种好用的交互图形程序器,用图解法收集设计数据,为模拟系统作准备,似乎是解决上述问题的良好途径。GTE AE实验室已经把图形程序成功地应用于电路设计的模拟中。图形程序器将要求设计者用图解法构思设计草图和编辑设计图案,当然要用所准备的实用程序。就是说设计者在完成其设计草图后,可以用实用程序给下列模拟系统提出所需要的设计信息。譬如Norwegian电子工业的CAD/CAM(计算机辅助设计/计算机辅助测量)的CASS用图解法集中设计信息,然后通过TEGAS输入转换实用程序把所判定了的信息传输到TEGAS模拟系统里。

设计者的性质特征

人的因素是关系到设计者能否接收应用交互系统的首要关键,人们的基本性能特征是已经确定的,下面所举的一些特征直接和交互系统相关。

①,人们在头脑里保持很长的数据串的记忆是困难的;如果出现中断,要记住键盘的长顺序

也是不容易的。

②, 设计者在等待结果的长时间中, 很容易不耐烦, 那么这也是设计者疲劳的原因, 并且降低了设计者的能动性。

③, 即使是低速的例行应答也将会引起不稳定性, 这对设计者是有影响的。

④, 设计人员用图形模式表示比用数据串表示更容易理解。

⑤, 设计人员更易于接收熟悉和始终不变术语。

问题说明

由于交互图形处理器主要的问题是把这种简单方式用于作设计图, 并给下列的模拟系统收集所需要的信息, 要达到此目的, 必须解决以下问题:

①, 要了解设计者的工作特征和心理状态, 提供一种容易学习, 容易使用的图形输入方式, 以方便设计者的作图, 编辑图形。

②, 编辑功能的功效要足以满足用户的需要。

③, 设计图信息必须适当地贮存起来, 以便用户可以随时编辑, 那就是说必须提供重新运行的能力。

④, 所贮存的设计图信息处理要简单, 以利于模拟系统的输入。

⑤, 必须给设计图信息提供灵活的数据结构, 因为图形处理器不是只限于一个模拟系统。设计就是综合其一系列已知条件完成所要求达到的功能过程, 在设计人员的作图过程中, 不要因图形最后一个元素的解决而受其干扰。而以下问题也是应解决的。

⑥, 元素的图形表示和符号组成也应适当地贮存, 以便于必要时恢复。

⑦, 设计图中的元素之间的关系也要调度得当。

要解决①和②的问题, 必须仔细地设计图形输入; 要解决③和⑦的问题, 必须把某些有关的数据库管理功能用到图形处理器上。在作图中或增加图中元素要提高灵巧度, 必须要把需要增加元素类的图形能力增加到图形数据库里。

系统综述

系统是在VAX11/780计算机上执行操作的, Tektronix存贮/刷新图形CRT是用于图形输入和输出的。用FORTRAN IV作为程序语言, 总的软件结构如图1所示。这个图形处理器目的在于收集一个以上的数字逻辑模拟系统的信息, 于是就要求数据结构的灵活性。在这个系统里, 用关系数据库支援(RDBS)就可以达到灵活的目的。而RDBS模块保持有数字逻辑元素的特性, 以及示意图关系, 这样只要把它们输入到以下的模拟系统中, 就可得到恢复和应用。

图形处理器的功能执行是由符号库模块, 示意图模块及RDBS接口模块完成的。符号库模块是用于符号库的符号构成; 示意图模块是用于示意图的作图; RDBS接口模块是用于保持符号关系(SYR), 示意图关系(SCR)以及网络关系(NWR)的。

符号库包含有SYR文件和基本图符文件, SYR文件贮存各个元素的一般特征, 而基本图符信息文件贮存基本元素的作图图符。设计者在用图形处理器作出设计图之前, 所需要的元素的信息一定要贮存在符号库里。设计者必须通过符号库模块把新的元素加到符号库里。在作示意图时, 就会形成两种关系, 如SCR和相关联的文件关系; NWR和相关联的文件。SCR贮存各个符号分布特征, 譬如各个符号分布的位置, 位移以及比例系数。NWR贮存符号分布的扇入扇出关系, 相关联文件把所有线构成的扇段都贮存在示意图里。

利用图形处理器，通过三种关系，二个文件，就可适当地控制设计信息，把设计图在图形终端显示出来。由于图形处理器的目的是给模拟系统收集信息，那么每个模拟系统都需要输入转换程序，如图2所示。我们执行操作的是用于DFSS—1模拟系统，即以三色调精确延时为基础的数字模拟系统。

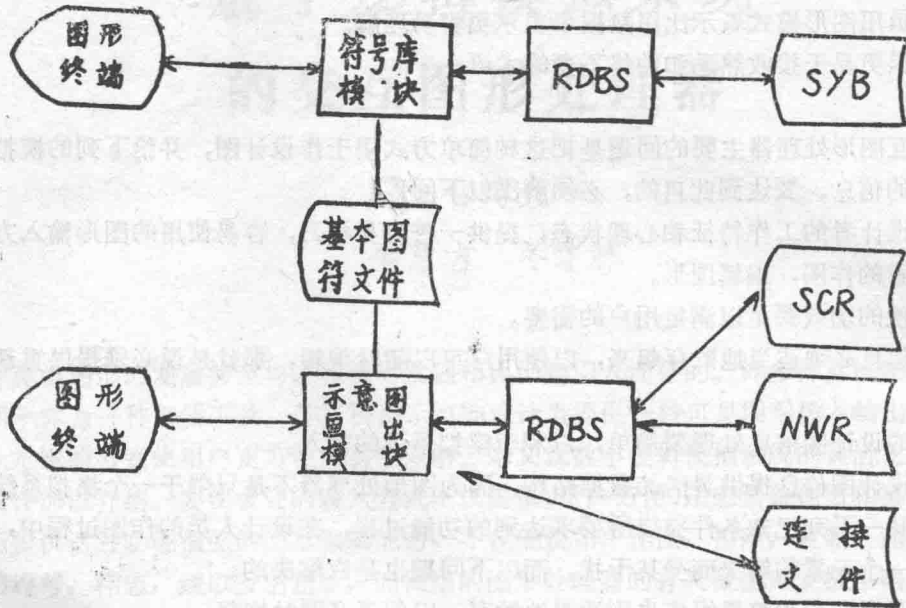


图1 总软件结构

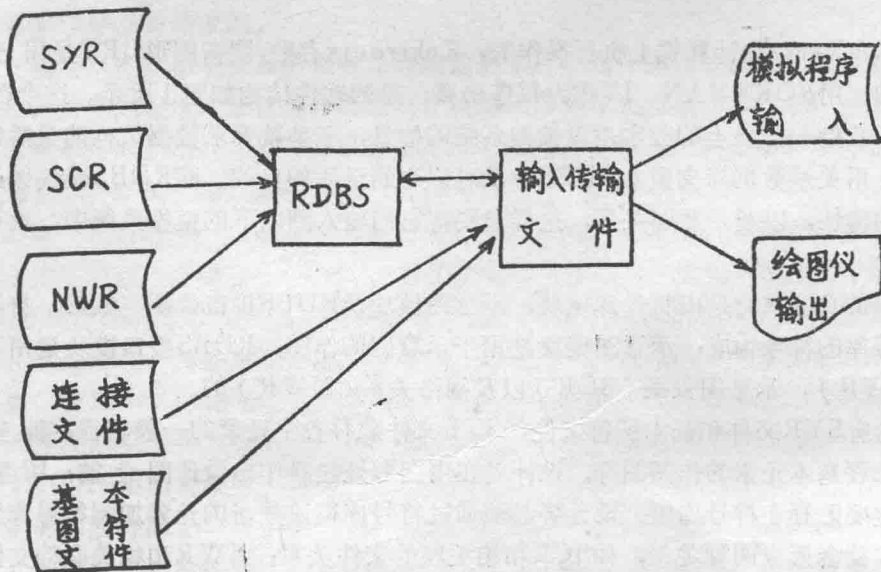


图2 所产生的各种关系与文件，以及输入传输程序之间的相互关系图

图形指令

设计者和图形处理器间的交互是通过设计者发出的一系列指令步骤完成的,设计者要得到合宜的交互,指令步骤一定要达到下列指标:

- ①,简单明了
- ②,容易使用
- ③,对用户的请求响应均匀
- ④,对于输入键敲击分配情况的考虑
- ⑤,纠错,避免用户不需要的“有效”指令的错误
- ⑥,用户可使任何未完成的指令在指令序列任意一步失灵。

图形指令是生成图形或处理图形的指令,从而可把它分成以下四份:

- ①,操作
- ②,操作对象
- ③,操作说明
- ④,终止作用

按光按钮就可启动操作部份,把屏幕的一部份示明为操作部份,可产生所需要的指令。操作的对象可以看成是符号、线,构成符号表示法,构成线的表示法,构成标记表示法或屏幕上信息区表示法。操作说明部份包括操作部份的附加信息。终止部份是键冲程“;”,指令语句如下所示:

操作对象,说明;

有些指令不需要操作对象和/或说明,比如: WINDOW 指令。

示意图模块上执行十种指令如下: 譬如

SYMBOL: 生成符号分配

WIRE: 生成线

LABEL: 把标记赋于符号或线

ERASE: 擦除符号、线、标记、或信息区

COPY: 复制信息区

MOVE: 把信息区移至另一个信息区

REPLACE: 置换符号

WINDOW: 改变观察窗口,改变观察窗口是示意图显示在屏幕上的部份。

HELP: 出示辅助信息,以帮助用户

RETURN: 返回调用模块

设计图的构成使指令程序化,为了减少设计者的输入键的敲击,除了刚开始按指令光按钮外,不按就可以启动同样操作的指令序列。

逻辑电路示意图的信息表示。示意图是由一组符号,连接线和标记构成。可以把符号和线看成是其特征的表示,那么就下列信息来表示示意图:

- ①,出现符号
- ②,出现符号的象征
- ③,符号间的相互连接
- ④,每条线的说明