



[12] 发明专利申请公开说明书

[21] 申请号 89109341.9

[51] Int. Cl.³
G06F 9/455

[43] 公开日 1990年8月1日

[22] 申请日 89.12.11

[30] 优先权

[32] 89.1.19 [33] US [31] 299,362

[71] 申请人 惠普公司

地址 美国加利福尼亚州

[72] 发明人 史蒂文·O·米勒

[74] 专利代理机构 中国专利代理有限公司

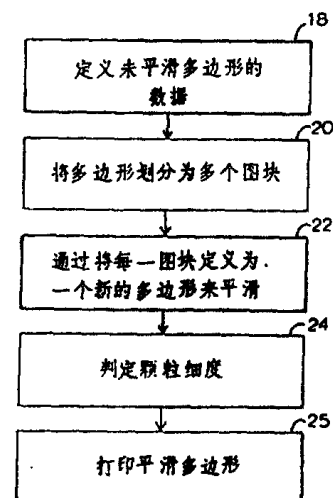
代理人 何耀煌 程天正

说明书页数: 6 附图页数: 3

[54] 发明名称 多边形平滑方法

[57] 摘要

本发明的多边形平滑方法包括以带有比原多边形更多顶点的另一多边形来描述原多边形的软件。通过确定为得到可接受输出质量所需的细度的步骤(34)以及通过用所需细度打印新的多边形的 Bezier 三次曲线的步骤(42)来构造新的多边形。



<39>

1. 一种使多边形平滑的方法，其特征在于包括：

将多边形划分为多个图块的步骤(20)，

通过将多边形的第一图块定义为新的多边形而使多边形的第一图块平滑的步骤(22)，

对该第一图块确定为得到可接受的输出质量所需的颗粒细度的步骤(24)，

对每个多边形图块重复所述平滑和确定步骤，以及

打印该平滑多边形的步骤(25)。

2. 一种由打印系统用来平滑并打印具有若干顶点的多边形的方法，其特征在于包括：

接收定义待平滑的多边形的数据的步骤(26)，

计算该系统所允许的最大颗粒细度的步骤(28)，

对多边形的每个三顶点集合计算 Bezier 曲线的控制点的步骤(30)，

确定所计算的控制点之间的线段长度的步骤(32)，

使用所确定的线段长度来计算为打印由所算出的控制点确定的 Bezier 曲线所需的最小颗粒细度的步骤(34)，

选择所允许最大颗粒细度和所需要最小颗粒细度中的较小者作为最终颗粒细度(36)的步骤，以及

打印具有最终颗粒细度的与所计算出的控制点对应的 Bezier 曲线的步骤(42)。

3. 权利要求 2 的方法，其特征在于：所述确定所计算的控制点之间线段长度的步骤包括对所计算控制点之间的像素个数进行计数。

4. 权利要求 3 的方法，其特征在于：所述使用所确定线段长度来计算所需最小颗粒细度的步骤包括该线段除以 4 个像素并对商舍入而取最接近的整数。

多边形平滑方法

本发明涉及使多边形平滑的方法，更准确地说，本发明涉及通过将多边形划分为多个图块并使每个这种图块平滑化来使多边形平滑的软件。

计算机图形大部分由多边形构成。多边形是由端端连接的直线所组成的图形。两线的连接点称为顶点。封闭多边形是其中所有直线端点都连接并形成“封闭”回路的多边形。封闭多边形的顶点可当作是封闭的。封闭多边形的一个例子是五边形。开放多边形是其中直线未连接形成封闭回路而有两“开放”端的多边形。开放多边形的一个例子是“V”。开放多边形的起始和终止点可称作顶点或开放顶点。

由于计算机图形图象大部分是由多种多边形构成，所以图象常常有不希望有的粗糙或齿状的表面。例如，计算机用户可能希望产生带有弧线体形状的汽车的形象。由于计算机图形是用多边形打印的，所以，轿车车体由连接一起的直线组成。线连接产生粗糙或齿状的表面而不是所要求的平滑曲线。因此，已研究一些方法用来使这样的多边形平滑化。

将多边形平滑化指的是修改多边形的粗糙或齿状的表面，使之呈现待产生的平滑曲线。过去，一直是通过使用一般称作Bezier三次曲线算法的算法来执行这种平滑过程。Bezier算法在计算机图形图象方面的一种应用在“Postscript Language Reference Manual”

(Adobe Systems, Inc., Copyright 1985)中进行了讨论。

本发明的方法通过使用Bezier算法连同确定图形图象所需“平滑”或“精细”程度的特征使多边形平滑，然后打印该结果。通过确定所需精细度，本发明的方法使图形图象质量最优而使平滑图象所需时间及计算机存储器最少。

本发明的多边形平滑方法包括将一多边形描述为带有比原多边形有较多顶点的另一多边形的软件。通过确定得到可接受的输出质量所需细度以及通过用所需细度打印新多边形的Bezier三次曲线来构造所述的新多边形。

图1展示计算机产生的未平滑的图形图象。

图2展示按照本发明的方法，计算机所产生的已平滑的图形图象。

图3为本发明多边形平滑方法的框图。

图4为本发明方法的详细框图。

图1示出了标号10的计算机产生的汽车的图形图象。该图象也可称为封闭多边形。该图象的外表面12粗糙并且显出产生该多边形的相连接的直线。

图2示出了使用本发明的多边形平滑方法由计算机产生的相同的图形图象14。外表面16也由连在一起的线构成，但是，这些线已平滑化，以产生得到所要求质量所需的细度。

图3是展示本发明多边形平滑方法的方框图。首先，在步骤18，收集定义未平滑多边形的数据，并将该多边形划分为多个图块，如框20所示。多边形图块是由预定个数的顶点所定义的该多边形的一部分。例如，在最佳实施例中，每三个顺序的顶点一组构成一多边形图块。从该定义来看，很显然多边形图块可以重叠。

然后，本方法通过将图块定义为顶点数增加的新的多边形来平滑每一个图块，如步骤22所示。然后，连接新多边形以形成“平滑的”多边形。形成该平滑多边形的线较短，所以外表面的粗糙并不显著。

在步骤24，确定该新多边形的达到可接收输出质量所需的细度。由于精细程度规定被打印的图形具有多大“颗粒度”，所以，常常将所述细度称为“粒度”或“颗粒细度”。简单地说，该颗粒细度是将构成平滑多边形的直线划分成多个较短线、以便取得可接受的打印质量的分度。该细度规定应将平滑多边形划分为多少条较短的线。实验表明，将平滑多边形的每一条线除以 $4/300$ 英寸，取最接近的整数，然后，打印具有所确定的颗粒细度的平滑多边形的各线，就能得到可接受的输出质量。在这个例子中， $4/300$ 英寸被认为是4个像素，其中，该图象以300像素/英寸的密度进行打印。可以看到，线越短，颗粒细度越小。这样，本发明的方法对每一线确定不同的细度，因此只有在需要时才使用较大的细度。

然后用所确定的颗粒细度来打印平滑多边形，如框25所示。

图4是本发明的多边形平滑方法的详细框图。首先收集定义未平滑多边形的数据，如框26所示。

使用本发明方法的每个系统都会有诸如有效存储器的系统限制。例如，某计算机系统可能有其中可描述多边形的32K字节的有效存储器。这种存储器局限由于限制了未平滑多边形可划分的线段个数数据而限制了多边形的平滑程度。因此，步骤28识别该系统所允许的最大颗粒细度，即，换言之，每一线段最大的分段数目。又，该极限值随存储器而变。

步骤30对未平滑多边形每三个顶点一组计算Bezier控制点而将

未平滑多边形划分多个图块。具体地说，本发明的方法识别未平滑多边形的前三个这样的顶点 P_0 、 P_1 和 P_2 。Bezier曲线的控制点可标识为 X_0 、 X_1 、 X_2 和 X_3 。封闭顶点组的控制点可按以下等式计算：

$$X_0 = (P_0 + P_1) / 2$$

$$X_1 = 1/6 P_0 + 5/6 P_1$$

$$X_2 = 5/6 P_1 + 1/6 P_2$$

$$X_3 = (P_1 + P_2) / 2$$

对既有开放顶点又有封闭顶点、而且第一顶点 P_0 是开放的顶点集，那么按以下等式计算控制点：

$$X_0 = P_0$$

$$X_1 = 1/3 P_0 + 2/3 P_1$$

$$X_2 = 5/6 P_1 + 1/6 P_2$$

$$X_3 = (P_1 + P_2) / 2$$

对既有开放又有封闭顶点、而且第一顶点 P_0 为封闭的顶点集，控制点计算如下：

$$X_0 = (P_0 + P_1) / 2$$

$$X_1 = 1/6 P_0 + 5/6 P_1$$

$$X_2 = 2/3 P_1 + 1/3 P_2$$

$$X_3 = P_2$$

对以开放顶点开始和结束的三顶点集合的控制点，按以下等式计算：

$$X_0 = P_0$$

$$X_1 = 1/3 P_0 + 2/3 P_1$$

$$X_2 = 2/3 P_1 + 1/3 P_2$$

$$X_3 = P_2$$

这些控制点表示 X轴上的坐标值。类似地，可用相同等式——用Y替代每个X，来得到Y轴的坐标值。

对未平滑多边形每个三顶点集合计算其控制点、并存储于存储器中，用来计算Bezier曲线。对每个三顶点P0、P1和P3的集合，计算四个控制点。下一个三顶点P1、P2、P3集合的第一控制点为前一集合的最后一个控制点。这样，控制点不重叠。

控制点的计算也将未平滑多边形划分为多个线段。顺序的控制点之间的直线构成这些线段。然后，步骤32确定每一条这种线段的长度。在最佳实施例中，按其密度为每英寸 300个像素的像素的数目来量度所述长度。

步骤34确定打印具有所要求的平滑度、由控制点所限定的Bezier曲线所必须的最小颗粒细度。通过将每个线段长度除以预定分解数来计算出最小颗粒细度。在最佳实施例中，线段长度除以 4个像素，其中，有每英寸 300像素。将所得到的商舍入而取最接近的整数，该整数构成所需最小颗粒细度。又，简单地说，颗粒细度就是将线段划分为许多更短的线以致在打印时得到可接受的输出质量的分度。该线段越短，颗粒细度越小。而颗粒细度受到描述更短线段所必须的存储器的限制。

接着，本发明的方法判定所需最小颗粒细度是否小于或等于允许颗粒细度，如步骤36所示。如果所需最小允许颗粒细度大于最大允许颗粒细度，那么，该系统不能打印具有该最小颗粒的多边形，所以，最终颗粒细度选择为最大允许颗粒细度，如步骤38所示。如果所必需的最小颗粒细度小于或等于最大允许颗粒细度，那么，本系统能够用最小颗粒细度作为最终粒度，如框40所示。

然后，步骤42打印带有所述最终粒度的Bezier曲线段。该Bezier

曲线段是每个三顶点集合的、由Bezier算法定义的曲线的一部分。在以下三个方程中使用计算出的控制点来计算X轴的常数 A_x 、 B_x 和 C_x 以及相对应的Y轴常数 A_y 、 B_y 和 C_y （未示出）： $X_1 = X_0 + C_x / 3$
 $X_2 = X_1 + (C_x + B_x) / 3$

$$X_3 = X_0 + C_x + B_x + A_x$$

然后在以下Bezier方程中使用这些常数来绘出Bezier曲线，其中， t 为定义曲线在其间延伸的距离的无量纲数，例如 $0 \leq t \leq 1$ ：

$$X_t = A_x t^3 + B_x t^2 + C_x t + X$$

$$Y_t = A_y t^3 + B_y t^2 + C_y t + Y$$

简言之，本发明的方法计算了未平滑多边形的每个三顶点集合的控制点。将这些控制点存储在存储器中并用来计算整个未平滑多边形的Bezier曲线。本方法还确定为打印平滑多边形所必需颗粒细度，使得该平滑多边形具有可接受的输出质量。当Bezier曲线和所必需最小颗粒细度都已计算出来时，本方法打印整个平滑多边形。

本发明的方法可用于打印计算机图形图象的系统中。本方法使这种图象平滑，以获得高质量输出，同时，提高处理速度以节省时间。

尽管文中已描述了本发明的最佳方式和最佳实施例，但仍可作变型和修改而不偏离本发明的精神。

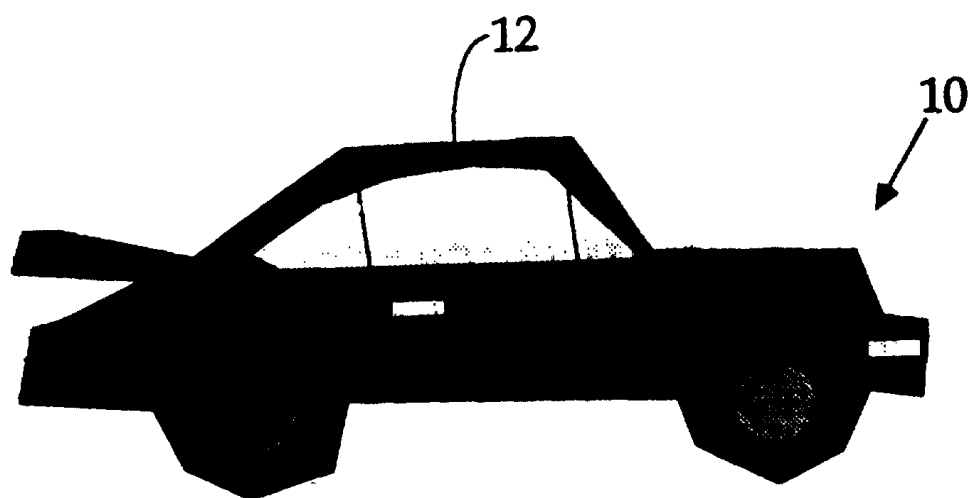


图 1

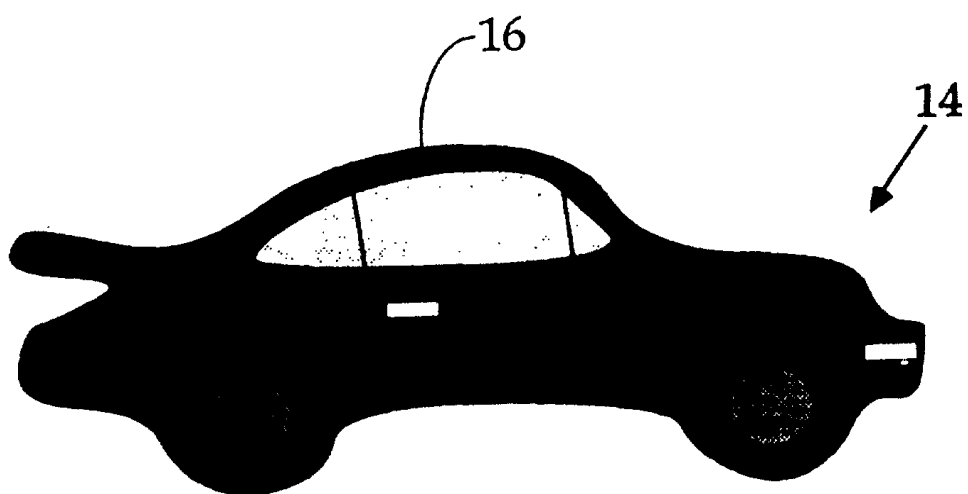


图 2

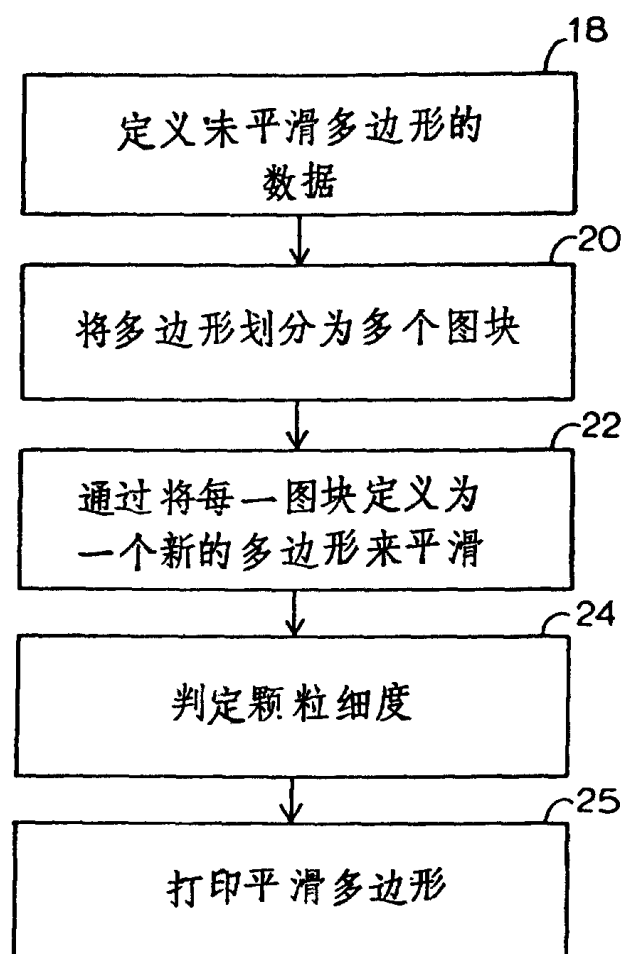


图 3

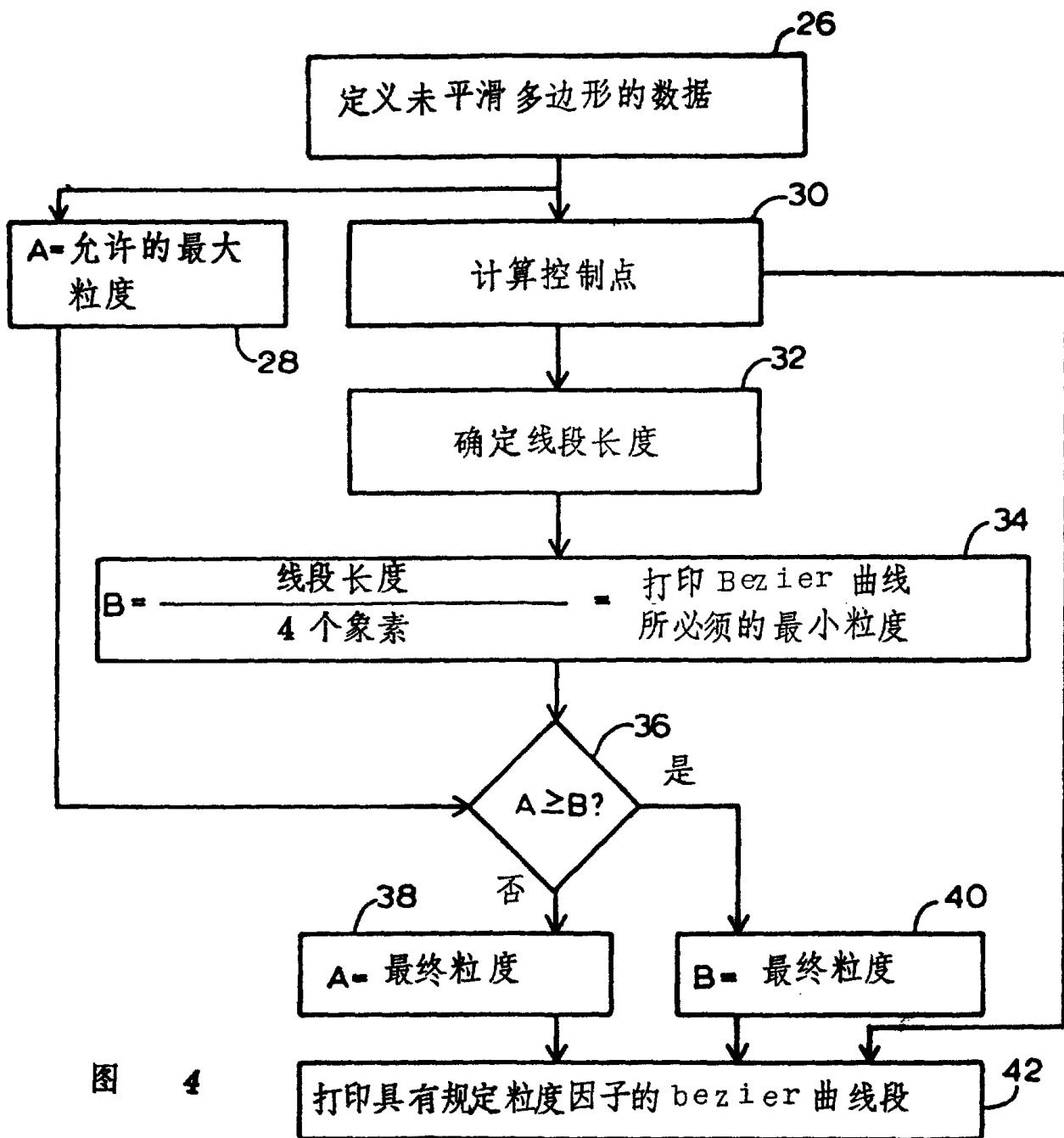


图 4