

文章编号:1007-2780(2016)04-0392-07

基于灰度控制的任意端点直线反走样算法研究

杨军锋¹,李乐意¹,杜 军¹,王海明²,杨 朴³

(1. 空军工程大学 航空航天工程学院, 陕西 西安 710038;

2. 陕西华燕航空仪表有限公司, 陕西 汉中 723102;

3. 陆军航空兵学院 飞行理论系, 北京 101123)

摘要:针对使用灰度控制算法不能准确绘制端点不在像素中心点的直线的问题,结合 Wu 算法,提出了一种基于灰度控制的任意端点直线反走样算法。介绍了 Wu 算法思想和灰度控制反走样算法,对两种算法进行数学分析,提出灰度控制不能解决的问题及问题产生的后果。结合 Wu 算法提出了新的基于灰度控制的算法,新算法减少了距离计算和灰度转换的计算,同时解决了任意端点直线的反走样问题。对新算法的效率和反走样效果进行仿真计算。仿真结果表明,新算法比 Wu 算法效率更高,绘制直线所用时间平均减少 33.91%;新算法比灰度控制算法绘制直线更准确,特别是在直线移动的过程中,有较好的动态显示效果。本文提出的新算法,效率较高,显示效果较好,具有很高的应用价值。

关 键 词:反走样;灰度控制;Wu 算法;任意端点直线

中图分类号:TP391.9 **文献标识码:**A **doi:**10.3788/YJYXS20163104.0392

Anti-aliasing algorithm for arbitrary-endpoint line based on gray control

YANG Jun-feng¹, LI Le-yi¹, DU Jun¹, WANG Hai-ming², YANG Pu³

(1. *Aeronautics And Astronautics Engineering College, Air Force Engineering University, Xi'an 710038, China;*

2. *Shaanxi Huayan Aero-instrument Co. Ltd., Hanzhong 723102, China;*

3. *Flight Theory Department, Army Aviation Institute, Beijing 101123, China)*

Abstract: Aiming at the problem that gray control algorithm can't draw accurately a line whose end-points do not in the center of a pixel, combined with the Wu algorithm, an anti-aliasing algorithm for arbitrary-endpoint line based on gray control is proposed. Firstly, Wu algorithm and gray control algorithm are introduced in the paper. Through analyzing the both algorithms, advantages and disadvantages of them are proposed. Then, combined with the Wu algorithm, a new algorithm based on gray control is put forward. The new algorithm reduces the calculation of distance and gray transformation, solves the anti-aliasing problem of arbitrary-endpoint line. Finally, the efficiency and anti-aliasing performance of the new algorithm is simulated. The simulation results show that the new algorithm has higher efficiency than the Wu algorithm, consuming time is reduced by an average of

收稿日期:2015-04-03; **修订日期:**2015-06-15.

基金项目:总装预研基金资助项目(No. 102080501);航空基金资助项目(No. 20100818015)

Supported by The General Armament Department Pre-Research Foundation(No. 102080501); Aeronautics Foundation(No. 20100818015)

* 通信联系人, E-mail: lileyilixun@126.com

33.91% when drawing a line, and is more accurate and has better anti-aliasing effect than the gray control algorithm. Especially in the process of line's moving, the algorithm has a good dynamic display effect. The new algorithm proposed in this paper has higher efficiency, better display effect and very high application value.

Key words: anti-aliasing; gray control; Wu algorithm; arbitrary-endpoint line

1 引 言

在嵌入式显示系统中,图形显示占据重要地位,由于直线是构成图形的基本图元,因此研究直线的绘制与显示有着非常重要的意义。光栅化图形显示时,利用离散的像素点表示连续直线,会出现明显的阶梯状或锯齿形的边缘,严重影响显示效果,这种用离散量表示连续量而引起的失真叫做走样。而减小或消除这种失真的技术叫做反走样^[1-2]。

反走样技术主要分为两类^[2-4],一类是提高光栅显示器的分辨率,另一类是使用反走样算法进行弥补。对于第一类,成本相对较高且只能减轻不能消除走样现象。反走样算法的核心思想都是通过直线边缘灰度的缓慢变化来消除走样现象。目前广泛采用的反走样算法主要有两种:区域采样和 Wu 算法^[5]。区域采样是将直线视作一个有一定宽度的矩形,计算矩形与像素的重叠面积,从而将像素赋予不同的灰度实现反走样^[6-7]。由于区域采样算法效果较好,被大量用于 3D 建模、图像的边缘反走样和罗盘仪表的反走样绘制等^[2,8-9]。但是,由于计算直线与像素点的重合面积涉及到大量浮点数计算,算法的缺点是计算比较复杂^[10-11]。Wu 算法将直线视为没有宽度的理想直线,计算一对像素中线与理想直线的距离,根据距离赋予一对像素不同灰度,来实现直线的反走样,计算相对简单^[12]。因此,Wu 算法在嵌入式设备中得到广泛应用^[13-15]。

传统的算法都是先计算重叠面积或距离,通过灰度转换来实现反走样,计算较为复杂。同时在工程应用特别是嵌入式系统对计算量小、动态效果好的反走样算法有很大的需求^[13-16]。在文献^[17]中提出一种基于灰度控制的反走样算法,减少了距离(或面积)的计算和灰度转换的计算,直接利用灰度控制生成反走样直线。但是该算法要求直线的起点和终点都在像素的中心点上,以致端点为小数的理想直线与反走样直线之间存在较大误差,导致直线在移动的过程之中产

生跳变、扭曲等现象。这种现象严重影响了直线的动态显示效果,在一些显示系统中是绝对不允许出现的,如机载显示设备等。

本文根据灰度控制反走样算法原理,结合 Wu 算法,提出了一种适用于任意端点直线的反走样算法。

2 Wu 算法和灰度控制算法

在直线反走样研究中,由于斜率 k 为 0、1、 ∞ 时,直线不存在走样现象,在研究反走样时通常只考虑 $0 < k < 1$ 时的反走样算法,其他斜率的直线可以通过 x,y 关系转化为 $0 < k < 1$ 的直线。为方便叙述以下仅讨论 $0 < k < 1$ 时的直线反走样算法。

2.1 Wu 算法基本原理

Wu 提出了简单有效地两点模式反走样算法。将沿长轴方向每前进一个像素单位(步进),在短轴方向同时点亮两个像素,且两者的亮度和为一个满亮度。根据理想直线到达像素中心点距离赋予像素不同的灰度。其基本原理如图 1 所示。图中每个小方格代表一个像素点,黑点为像素中心点,直线 L 为理想直线。设最大灰度为 G ,像素中心点的距离为 1。

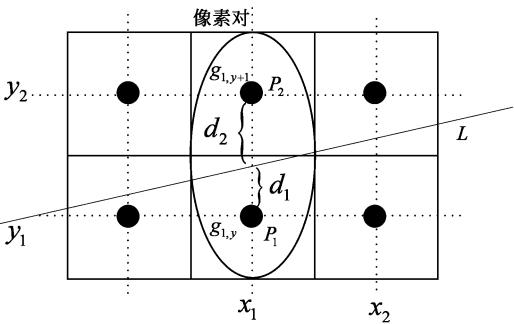


图 1 Wu 算法原理图

Fig. 1 Schematic of Wu algorithm

图 1 中 P_1 和 P_2 分别表示一对像素的中心点, d_1 和 d_2 分别是 P_1 和 P_2 距离理想直线的距离, $d_1 < 1$ 时,根据 d_1 和 d_2 计算出像素灰度大小:

$$\begin{cases} g_{1,y}=d_1 \cdot G \\ g_{1,y+1}=G-g_{1,y} \end{cases} \quad (1)$$

若 $d_1>1$,Y 方向产生步进,即:

$$\begin{cases} y=y+1 \\ d_1=d_1-1 \\ g_{1,y}=d_1 \cdot G \\ g_{1,y+1}=G-g_{1,y} \end{cases} \quad (2)$$

式(1)、式(2)中 $g_{i,y}$ 与 $g_{i,y+1}$ 表示 X 为 x_i 时 (x,y) 和 $(x,y+1)$ 像素对的灰度值。

2.2 灰度控制算法基本原理

图 2 表示灰度控制反走样直线生成算法原理,在图 2(a)中表示理想直线在光栅显示器中的位置。根据传统的 Wu 反走样生成算法,生成的直线如图 2(b)所示。Wu 反走样直线生成过程中,X 方向上产生步进以后需要考虑的问题有两个:一是 Y 方向上哪一个像素对被选择,二是被选择的像素对的灰度分别为多少。而从灰度的角度来考虑直线的生成,则要更为简单。在图 2(b)中,发现像素对灰度值和被选择的像素对,有一定规律:在起点时,所对应的像素灰度为 0,随着 X

的步进,灰度不断增加,直至最终超过最大灰度,Y 方向产生步进。简而言之,反走样直线绘制时,应完成一次灰度循环后,产生 Y 方向的步进。利用这种规律,构建灰度控制反走样算法。

设最大灰度为 G ,在起始点 (x_1,y_1) 上其灰度为 0,其中 (x_1,y_1) 为整数坐标。X 步进时,灰度值不断增加:

$$\begin{cases} g_{i,y}=g_{i-1,y}+\Delta G \\ g_{i,y+1}=G-g_{i,y} \\ \Delta G=k \cdot G \end{cases} \quad (3)$$

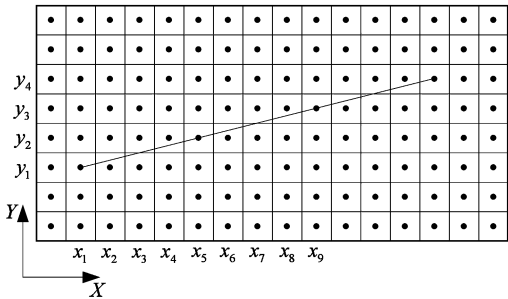
其中: ΔG 表示 X 步进时,像素对中 $(x+1,y)$ 相对 (x,y) 像素灰度增加值。 k 为理想直线的斜率。

但若灰度 $g_{i,y}>G$,则产生 Y 方向的步进,即:

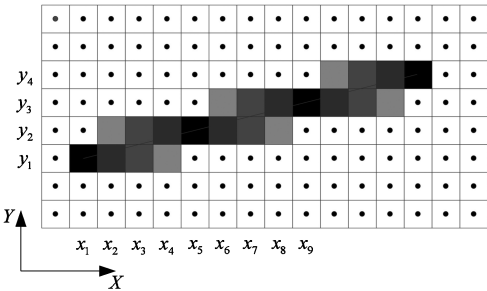
$$\begin{cases} y=y+1 \\ g_{i,y}=g_{i,y}-G \\ g_{i,y+1}=G-g_{i,y} \end{cases} \quad (4)$$

根据式(3)和式(4)可以构建一个简洁高效的灰度控制反走样算法,该算法减少了距离转化为灰度的计算,

虽然灰度控制反走样直线生成算法简洁高效,但是,它要求直线端点在像素中心上,即端点坐标都为整数。当端点坐标并不是整数时反走样直线不准确。如图 3 所示,理想直线为 L_1 起点和终点都不在像素中心点上,绘制反走样直线时,所绘制的反走样直线是根据 L_2 绘制的,即所绘制的直线其实是理想直线为 L_2 的反走样直线。这显然是不精确的,特别是在直线移动、旋转的过程中出现跳变、扭曲等现象。



(a)理想直线的位置
(a) Position of ideal line



(b)反走样直线
(b) Anti-aliasing line

图 2 灰度控制反走样直线生成原理
Fig. 2 Generating principle of gray control anti-aliasing line

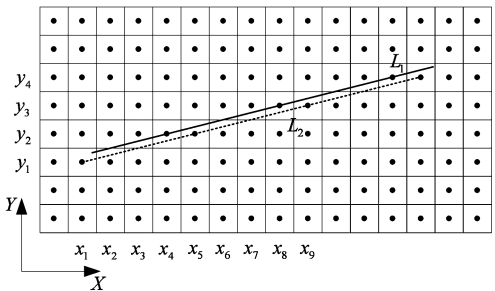


图 3 算法存在的问题
Fig. 3 Existence problem of the algorithm

因此,需要将灰度控制算法进行优化,将该算法推广至任意端点的直线,进一步满足任意端点直线的反走样需求。

3 新的直线反走样算法

如图 4 所示,一段直线起点为 (a,b) 终点为 (c,d) , a,b,c,d 为浮点数。 $x_0 < a < x_1, x_s < c < x_{s+1}$,称 x_0 上像素对为第 0 像素对, x_1 上像素对为第 1 像素对,依次类推,最后一个像素对为第 $s+1$ 像素对。

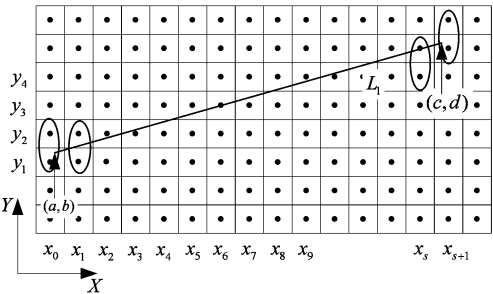


图 4 端点为小数的直线
Fig. 4 Line with decimal endpoints

3.1 第 1 像素灰度计算

首先,根据 Wu 算法思想,计算第 1 像素对的灰度值,如图 5 所示,只需计算出 d 的大小即能够求出 g_1 。

从起点 (a,b) 到第一点有 Y 方向可能出现两种情况:(a) Y 没有步进;(b) Y 产生步进。当 Y 没产生步进时,如图 5(a)所示,计算 d 。

$$d = (1 - \Delta a) \cdot k + \Delta b, \tag{5}$$

其中 $\Delta a, \Delta b$ 分别为 a, b 的小数部分。即:

$$\begin{cases} \Delta a = a - \text{floor}(a) \\ \Delta b = b - \text{floor}(b) \end{cases} \tag{6}$$

式(6)中 $\text{floor}(x)$ 表示比 x 小且距离 x 最近的整数。

在图 5(b)中, Y 产生了步进,计算 d :

$$\begin{cases} d = (1 - \Delta a) \cdot k + \Delta b - 1 \\ y = y + 1 \end{cases} \tag{7}$$

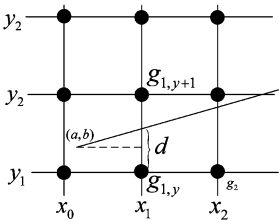
根据距离与灰度的关系可以求出第 1 像素对的灰度 g_1 。

$$\begin{cases} g_{1,y} = d \cdot G \\ g_{1,y+1} = G - g_{1,y} \end{cases} \tag{8}$$

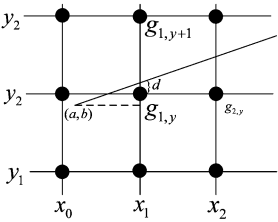
计算出 g_1 以后,就能够采用灰度控制算法绘出直线主体,这里的直线主体是指除第 0 和第 $s+1$ 像素对的直线。

3.2 直线主体的绘制

绘制直线的主体,即采用灰度控制的方法分



(a) Y 无步进情况
(a) Y has no step



(b) Y 有步进情况
(b) Y has a step

图 5 第 1 像素对灰度计算

Fig. 5 Computing of the first pixel pair gray scale

别求出 $g_i (2 \leq i \leq s)$ 。

$$\begin{cases} g_{i,y} = g_{i-1,y} + \Delta G \\ g_{i,y+1} = G - g_{i-1,y} \end{cases} \tag{9}$$

若 $g_i \geq G, Y$ 方向产生步进

$$\begin{cases} y = y + 1 \\ g_{i,y} = g_{i,y} - G \\ g_{i,y+1} = G - g_{i,y} \end{cases} \tag{10}$$

3.3 g_0, g_{s+1} 的计算

首先计算 g_0 ,若 $g_1 > \Delta G$,则 x_0 到 x_1 步进时 Y 方向无步进,设想直线 L 可以理解为直线 L' 向右平移得到的起点为 (a,b) 的直线,如图 6 所示,直线 L' 第 0 像素对灰度为

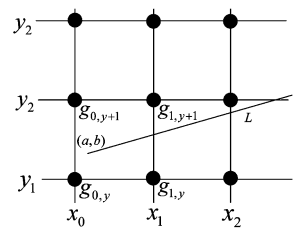
$$\begin{cases} g'_{0,y} = g_1 - \Delta G \\ g'_{0,y+1} = G - (g_1 - \Delta G) \end{cases} \tag{11}$$

同理,若 $g_1 < \Delta G$,则 x_0 到 x_1 步进时 Y 方向产生步进,即:

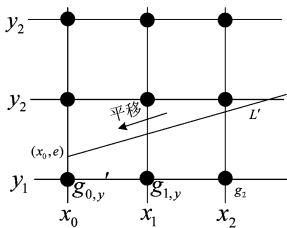
$$\begin{cases} g'_{0,y} = g_1 - \Delta G + G \\ g'_{0,y+1} = \Delta G - g_1 \end{cases} \tag{12}$$

当直线向右平移,此时由于起点 (a,b) 逐渐远离第 0 像素对,随着 Δa 的增大,第 0 像素对的两个像素的灰度应同时逐渐增大,这时像素对表现为“淡出”效果,像素对灰度和不再为 G , 由此得出:

$$\begin{cases} g_{0,y}=(G-g_{0,y})\cdot\Delta a+g'_{0,y} \\ g_{0,y+1}=(G-g'_{0,y+1})\cdot\Delta a+g'_{0,y+1} \end{cases} \quad (13)$$



(a)理想直线起点
(a)Starting point of ideal line



(b)理想直线的平移
(b)Shift of ideal line

图 6 第 0 像素对灰度值计算

Fig. 6 Computing of zeroth pixel pair gray scale

同理,求出 $s+1$ 像素对的灰度:

$$\begin{cases} g_{s+1,y}=(G-g'_{s+1,y})\cdot(1-\Delta c)+g'_{s+1,y} \\ g_{s+1,y+1}=(G-g'_{s+1,y+1})\cdot(1-\Delta c)+g'_{s+1,y+1} \end{cases} \quad (14)$$

在计算所有像素对的灰度之后,只需将不同像素赋予对应的灰度值,就完成了反走样直线的绘制。

4 仿真实验

本文基于灰度控制提出了任意端点直线反走样算法,为了验证算法的时间效率和准确性以及直线动态性能,在 Matlab 中进行如下仿真。

(1)首先,研究新算法的效率。利用 Matlab7.9.0 软件对比新算法与 Wu 算法的效率,采用的计算机 CPU 为 2.8G 主频的 Core i5 处理器,内存大小为 3G。分别用 Wu 算法和新算法绘制起点(1,1),X 方向跨度为 100 像素,斜率分别为 0.05、0.1、0.15……0.95 的直线。每条直线绘制 1 000 次,计算出每条直线所用的平均时间,如表 1 所示。

表 1 算法效率分析

Tab. 1 Analysis of efficiency the algorithm

直线斜率	Wu 算法/ ms	新算法/ ms	耗时减少/ %
0.05	0.426 683	0.437 534	-2.54
0.10	0.440 754	0.427 491	3.01
0.15	0.451 354	0.343 267	23.95
0.20	0.002 564	0.002 547	0.66
0.25	0.447 622	0.354 266	20.86
0.30	0.443 969	0.230 656	48.05
0.35	0.348 279	0.345 490	0.80
0.40	0.101 252	0.002 683	97.35
0.45	0.459 206	0.346 511	24.54
0.50	0.433 558	0.233 406	46.16
0.55	0.464 614	0.341 917	26.41
0.60	0.456 888	0.002 580	99.44
0.65	0.460 763	0.348 012	24.47
0.70	0.231 356	0.229 526	0.79
0.75	0.460 884	0.358 230	22.27
0.80	0.460 757	0.002 592	99.44
0.85	0.345 361	0.349 932	-1.32
0.90	0.474 773	0.238 535	49.76
0.95	0.465 722	0.356 746	23.40
耗时平均减少(%)			33.91

由表 1 结果可知,新算法在有些时候较 Wu 算法耗时没有明显的降低,在有些情况下,耗时减少极多。但在多数情况下,新算法要比 Wu 算法耗时减少 20%~50%,平均耗时减少 33.91%。说明新算法比 Wu 算法更有效率。

(2)仿真研究反走样直线的显示效果。仿真端点为小数时,直线的绘制效果,图 7 中表示直线起点(101,101.2)到终点(105,131.2)的直线分别在灰度控制算法、Wu 算法和新算法下生成的反走样直线。

图 7(a)采用灰度控制算法,起点和终点被近似为整数,因此绘制的直线并不精确。图 7(b)和图 7(c)中,反走样效果相对较好同时更准确,可看出采用 Wu 算法和新算法直线更准确,都有较

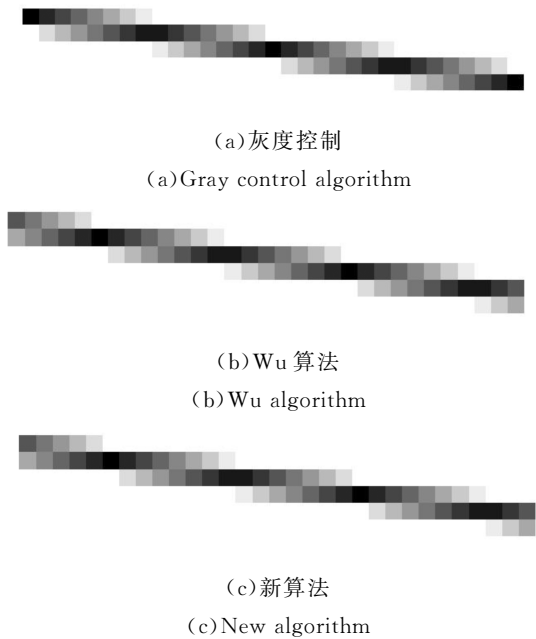


图 7 新算法与灰度控制算法、Wu 算法显示效果对比

Fig. 7 Display effect of gray control algorithm, Wu algorithm and the new algorithm

好的显示效果。

(3)仿真研究新算法的动态显示效果。在 MATLAB 中,使用新算法绘制一条逐渐移动的直线。仿真中使用斜率为 0.2,起点(1,1)的直线,每 0.1 s 直线向右平移 0.1 像素,仿真结果发现,直线的端点具有“淡入”、“淡出”的特性,克服了以往反走样直线的跳变、扭曲现象。动态效果显著提高,具有较高的应用价值。在直线平移的过程中,起点灰度变化如图 8 所示。

从图 8 中可以看出随着 Δa 增大,第 0 像素对灰度逐渐增大,其他像素对的灰度没有变化。说明采用新算法,有效地平滑了端点为浮点数的直线的端点。

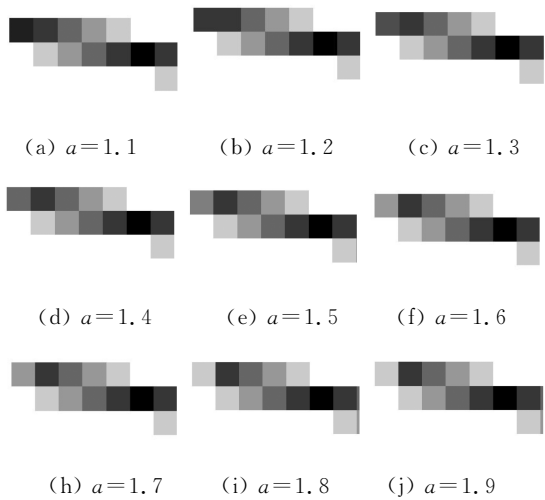


图 8 直线平移过程中起点灰度变化

Fig. 8 Gray change when the line shifting

5 结 语

本文针对反走样算法进行研究,指出了 Wu 算法和灰度控制算法的优缺点。通过结合 Wu 算法和灰度控制算法的优缺点。通过结合 Wu 算法,在灰度控制反走样算法的基础之上,提出了一种适用于任意端点直线的灰度控制反走样算法。与 Wu 算法相比,新算法减少了距离和灰度转换的计算,有效提高了算法的效率。仿真结果表明,采用新算法比采用 Wu 算法绘制直线所用时间减少 20%~50%,平均减少 33.91%。同时有效解决了灰度控制算法绘制直线的不精确等问题,解决了直线移动过程中的跳变扭曲等问题,取得了较好的动态显示效果。由于新算法具有计算效率较高、动态效果好等特点,因此如果将新算法运用于嵌入式设备的显示如罗盘仪表盘和地图可视化等方面,将能大大缩短直线绘制时间,提高实时性和准确性。

参 考 文 献:

[1] HEARN D, PAULINE B M. 计算机图形学[M]. 蔡士杰译. 第 3 版. 北京:电子工业出版社,2005.
HEARN D, PAULINE B M. *Computer Graphics with OpenGL*[M]. CAI S H, trans. 3rd ed. Beijing: House of Electronics Industry, 2005. (in Chinese)

[2] JIANG X D, SHENG B, LIN W Y, *et al.* Image anti-aliasing techniques for internet visual media processing: a review[J]. *Journal of Zhejiang University-Science C*, 2014, 15(9): 717-728.

[3] 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[J]. 计算机辅助设计与图形学学报,2003,15(1):71-75.
ZHANG B, ZHANG H C, JING Y Z. Improvement of high speed anti-aliasing algorithm for the display of compass

- [J]. *Journal of Computer Aided Design & Computer Graphics*, 2003, 15(1): 71-75. (in Chinese)
- [4] 石娟. 基于 OpenGL 的 VxWorks 系统反走样技术研究与应用[D]. 南京: 南京理工大学, 2013.
SHI J. Based on the embedded system OpenGL anti-aliasing technology research and application[D]. Nanjing: Nanjing University of Science and Technology, 2013. (in Chinese)
- [5] 汪丽娜. 飞机座舱显示系统仪表画面的反走样技术研究[D]. 长春: 吉林大学, 2007.
WANG L N. Research on antialiasing technology for instrument picture of aircraft cockpit display system[D]. Changchun: Jilin University, 2007. (in Chinese)
- [6] 毛峡, 刘运龙, 薛雨丽. 基于未加权区域采样的直线反走样算法[J]. 北京航空航天大学学报, 2013, 39(6): 717-722.
MAO X, LIU Y L, XUE Y L. New anti-aliasing straight line drawing algorithm based on unweighted area sampling[J]. *Journal of Beijing University of Aeronautics and Astronautics*, 2013, 39(6): 717-722. (in Chinese)
- [7] 刘涛, 高青伟, 赵国荣. 一种新的基于圆锥滤波的直线反走样生成技术[J]. 工程图学学报, 2006, 27(3): 61-64.
LIU T, GAO Q W, ZHAO G R. A new antialiasing technique for drawing straight line based on cone filter[J]. *Journal of Engineering Graphics*, 2006, 27(3): 61-64. (in Chinese)
- [8] LI C G, ZHANG Q R, ZHANG W Y. Antialiasing compass meter display by Gaussian integration[J]. *Computer Aided Drafting, Design and Manufacturing*, 2005, 15(1): 41-45.
- [9] JIMENEZ J, ECHEVARRIA J I, SOUSA T, *et al.* SMAA: Enhanced sub pixel morphological antialiasing[J]. *Computer Graphics Forum*, 2012, 31(2): 355-364.
- [10] 徐小良, 洪波. 一种基于整数坐标的亚像素精度区域采样反走样算法[J]. 中国图象图形学报, 2009, 14(12): 2438-2442.
XU X L, HONG B. A sub-pixel regional sampling anti-aliasing algorithm based on integer coordinate[J]. *Journal of Image and Graphics*, 2009, 14(12): 2438-2442. (in Chinese)
- [11] 张慧慧. 基于反走样算法和滤波技术的地图可视化研究[J]. 黑龙江大学自然科学学报, 2014, 31(5): 687-690.
ZHANG H H. The map visualization study of anti-aliasing algorithm and filtering techniques[J]. *Journal of Natural Science of Heilongjiang University*, 2014, 31(5): 687-690. (in Chinese)
- [12] WU X L. An efficient anti aliasing technique[J]. *ACM SIGGRAPH Computer Graphics*, 1991, 25(4): 143-152.
- [13] 李铂, 周建江, 夏伟杰, 等. 小角度直线反走样的改进 Wu 算法[J]. 液晶与显示, 2014, 29(4): 605-610.
LI B, ZHOU J J, XIA W J, *et al.* Modified anti-aliasing algorithm for small-angle line based on Wu algorithm[J]. *Chinese Journal of Liquid Crystals and Displays*, 2014, 29(4): 605-610. (in Chinese)
- [14] 吴连慧, 周建江, 夏伟杰, 等. 基于 Wu 反走样的三角形光栅化边缘反走样算法[J]. 液晶与显示, 2015, 30(1): 163-169.
WU L H, ZHOU J J, XIA W J, *et al.* Edge anti-aliasing algorithm of triangle rasterization based on Wu algorithm[J]. *Chinese Journal of Liquid Crystals and Displays*, 2015, 30(1): 163-169. (in Chinese)
- [15] 常进, 何援军, 田海山. 基于 OpenGL 的机载图形生成算法[J]. 工程图学学报, 2008, 29(3): 50-55.
CHANG J, HE Y J, TIAN H S. Graphics generation algorithm for aircraft platform based on OpenGL[J]. *Journal of Engineering Graphics*, 2008, 29(3): 50-55. (in Chinese)
- [16] 秦立红, 肖铁军, 华纯. 任意波形的彩色反走样算法研究与实现[J]. 计算机工程与设计, 2011, 32(11): 3784-3788.
QIN L H, XIAO T J, HUA C. Research and implementation of color anti-aliasing algorithm for arbitrary waveform[J]. *Computer Engineering and Design*, 2011, 32(11): 3784-3788. (in Chinese)
- [17] 牛连强, 张胜男, 钟玲. 反走样直线的灰度循环生成算法[J]. 中国图象图形学报, 2013, 18(1): 115-122.
NIU L Q, ZHANG S N, ZHONG L. Integral algorithm for generating anti-aliased straight line controlled by gray iteration[J]. *Journal of Image and Graphics*, 2013, 18(1): 115-122. (in Chinese)

作者简介: 杨军锋(1958—), 男, 陕西泾阳人, 副教授, 硕士生导师, 主要从事嵌入式系统、自动化仪表等方面的研究。

E-mail: 1464843009@qq.com

李乐意(1989—), 男, 河南夏邑人, 硕士研究生, 主要研究方向为: 嵌入式系统设计。E-mail: lileylilixun@126.com