

基于 OpenGL 的雷达显示器计算机仿真的实现

The Realization of Computer Simulation for Radar Indicator Based on OpenGL

王 宏 (桂林空军学院 广西 桂林 541003)
 樊世友 (军械工程学院 河北 石家庄 050003)
 陈再旺 莫裕孟 (桂林空军学院 广西 桂林 541003)

摘要:基于建模工具 GL Studio 和三维图形标准 OpenGL 对雷达显示器的计算机仿真进行研究。本文着重论述利用 OpenGL 对雷达显示器余辉现象和目标航路模拟的实现。在分析 RGB、HIS 颜色模型的基础上,建立了余辉现象及目标航路数学模型,并以某型雷达 P 型显示器的余辉显示为例给出了在 OpenGL 下进行模拟的具体算法。

关键词:OpenGL 雷达显示器 计算机模拟

雷达显示器是雷达显示目标的重要手段,无论是操作训练还是维修训练,显示器都是操作人员关注的焦点。所以雷达显示器仿真的效果对整个雷达仿真的效果具有很大影响。GL Studio 是一种全新模式的视景仿真建模工具,它直接生成基于 OpenGL 的 C++ 源代码,它在对复杂的仪器面板如飞机驾驶舱内的操作面板的建模、雷达面板的建模等方面有很大的优越性。OpenGL 是 SGI 公司开发的可独立于操作系统和硬件环境的三维图形库,由于其强大的图形功能和跨平台能力,已成为事实上的三维图形工业标准。在科学可视化、CAD/CAM、模拟仿真等领域得到了广泛的应用。本文以“通用雷达装备虚拟维修训练系统”为应用背景,借助 GL Studio 和 OpenGL 实现了对整个雷达装备的计算机仿真。在雷达显示器的仿真过程中,主要包括目标显示、余辉现象模拟和干扰现象模拟。这里着重介绍基于 OpenGL 的目标显示和余辉现象模拟的实现。

1 颜色模型的选取

雷达显示器大都使用单色阴极射线管 CRT (cathode-ray tube), 主要给用户显示的是亮度的变化,而计算机对颜色的管理一般都使用 RGB 模型。所以,必须首先使用计算机实现对显示亮度的处理。

RGB 模型在颜色空间是一个正立方体。该立方体

面和内部的任意一点颜色都可以由 R、G、B 三种颜色通过叠加得到。RGB 模型是面向硬件的。它的特点是便于硬件设计,但与人眼的直接感觉不直接对应。所以难以进行与视觉直接相关的效果处理,如对亮度的处理等。所以人们还经常使用别的颜色模型来对颜色进行描述,常见的有 CMY、HSV 等。本文使用基于视觉原理的颜色模型 HIS, 它由色调 H、亮度 I、饱和度 S, 在一个三棱锥空间定义颜色,如图 1 所示。色调由与颜色对应的主波长决定,是一种颜色区别于另一种颜色的要素;亮度是颜色的明暗特性;饱和度决定于对应色调的纯度,也就是“白色”的程度。

为了更方便地研究模型,我们还采用了用以匹配可见光的 CIE 色度图来定义颜色域。并将 RGB 三角形从 CIE 色度图中提取出来,脱离原来的坐标系,形成图 2 所示的色度坐标三角形。

由于计算机使用的是 RGB 颜色模型,而我们要描述的视觉效果要使用 HIS 模型,所以要在两个模型间建立变换。RGB 颜色空间到粗略 HIS 颜色空间的转换采用公式(1)来完成:

$$\begin{cases} I = (R+G+B)/3 \\ S = 1 - \frac{3 \cdot \min(R, G, B)}{R+G+B} \\ H = \begin{cases} \arccos\left\{\frac{[(R-G) + (R-B)]/2}{[(R-G)^2 + (R-B)(G-B)]^{1/2}}\right\} & G \geq B \\ 360 - \arccos\left\{\frac{[(R-G) + (R-B)]/2}{[(R-G)^2 + (R-B)(G-B)]^{1/2}}\right\} & G < B \end{cases} \end{cases} \quad (1)$$

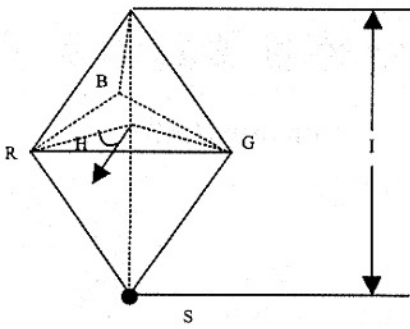


图 1 HIS 颜色空间

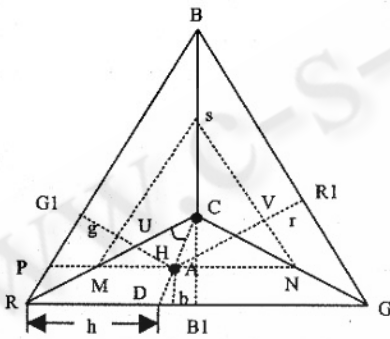


图 2 色度坐标三角形

2 CRT 长余辉模型

我们知道,雷达的显示器使用的是示波管,在它的前方有一个荧光屏。荧光屏担负着显示的任务。荧光屏使用如硫化锌、硅酸锌之类的荧光质,这些物质,在高速电子束的轰击下能发出可见光。

当电子束停止对荧光物质的轰击后,荧光质的发光仍能持续一定的时间才消失的现象叫余辉。一般以电子束停止轰击后,亮度由最大值下降到它的 2-5% 时所需要的时间定义为余辉时间。余辉时间可以分为:

短余辉:余辉时间为 1-10 微秒;

中余辉:余辉时间为 1-100 毫秒;

长余辉:余辉时间为大于 1 秒。

相关资料表明,荧光亮度一次指数衰减曲线方程可用式(2)来表示:

$$I = I_0 e^{-t/K_1} \quad (2)$$

其中 I 为余辉亮度, I_0 为涂层亮度, K_1 为时间衰减常数, t 为衰减时间。对于每种荧光质, I_0 和 K_1 均为常数。 I_0 越大,荧光衰减曲线越平坦,而 K_1 越大则余辉时间越长。下面我们以某型雷达的显示器为例确定上式中的参数。

首先,这里的余辉亮度 I 可以使用以上所论述的 HIS 颜色空间的 I 来描述。我们先采集一张雷达显示器余辉最亮时的照片,然后通过 PhotoShop 软件获得余辉的 RGB 值。再根据 RGB 颜色空间到 HIS 颜色空间的变换公式就可以得到 I_0 。以某型雷达的显示器为例,我们得到的 RGB 值为:

$$R = 255, G = 255, B = 0;$$

代入到式(1),可得:

$$H = 60, I = 166.7, S = 1;$$

于是我们就得到了 $I_0 = 166.7$ 。

然后,求第二个参数 K_1 ,经测量可以得到余辉的长度大约是 2 秒钟,于是有:

$$e^{-t} = 0.02$$

于是可得: $K_1 = 0.5$

这样,我们就得到了在 HIS 颜色空间中,某型雷达显示器的余辉数学模型为:

$$\begin{cases} I = 166.7 e^{-t} \\ S = 1 \\ H = 60 \end{cases} \quad (t > 0) \quad (3)$$

3 目标航路生成模型

计算机能够模拟目标航路是因为人们对目标运动的研究成果。通常情况下,目标的运动有匀速直线运动、匀加速直线运动、水平匀速圆弧运动等假定。本文以匀加速直线运动为例说明目标的生成。

事实上一个完整的高炮防空系统工作过程是这样的:雷达利用电磁波等方式,发现并跟踪目标,并且目标的有关参数传递给火控计算机,由火控计算机计算出高炮的射击诸元并传给高炮,然后由高炮射击目标。在这个过程中,雷达需要传给火控计算机的目标参数一般有目标的方位角(β),高低角(ϵ)和斜距离(D)。一般雷达也将目标的这些信息显示给雷达操作手。下面研究这三个量的生成问题。

当目标做匀加速直线运动时,如图 3 所示。假设

D_0 为某一参考时刻目标的斜距离, A 为目标的加速度, 目标航程为 s_0 , 航向角 Q , 航路捷径 d_1 , 初速 v_0 , 向下俯角 λ 都是已经假定好的, 于是, 在图 3 中, 我们可以得到:

$$\begin{cases} H(t) = H_0 - (v_0 t + \frac{1}{2} A t^2) \sin \lambda \\ s(t) = s_0 - (v_0 t + \frac{1}{2} A t^2) \cos \lambda \\ d(t) = \sqrt{d_1^2 + s(t)^2} \end{cases} \quad (4)$$

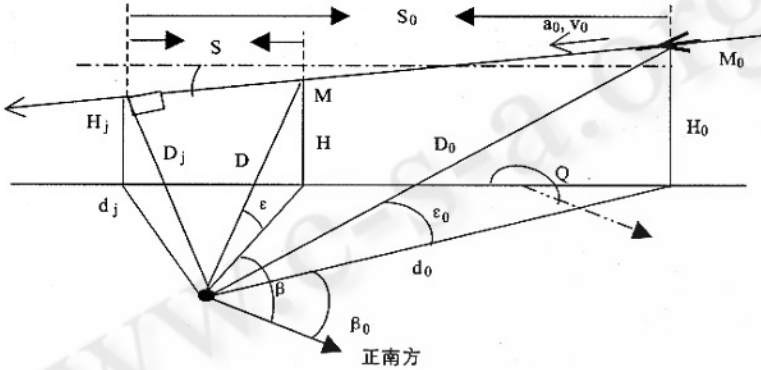


图 3 目标匀加速直线运动航迹

因此, 所需要的三个参数就可以按下面的式子计算生成。

$$\begin{cases} \beta(t) = Q + (-1)^k [\frac{\pi}{2} + \arctg \frac{s(t)}{d_1}]; k = \begin{cases} 1 & \beta \text{ 增加时} \\ 0 & \beta \text{ 减少时} \end{cases} \\ \epsilon(t) = \arctg \frac{H(t)}{d(t)} \\ D(t) = \sqrt{d(t)^2 + H(t)^2} \end{cases} \quad (5)$$

4 基于 OpenGL 的计算机模拟雷达显示器

在以上对余辉模型和航路生成分析的基础上, 就可以利用计算机模拟雷达显示器了。这里以某型雷达的 P 型显示器为例从具体实现的算法上进行论述。

P 型显示器进行工作时, 主要分为三个过程: 上电过程、扫描过程和断电过程。在这三个过程中都存在有亮度的变化。上电过程有扫描线渐亮的过程, 扫描过程时有长余辉现象, 断电过程有扫描线渐暗过程。

由于整个系统的视景都是在 OpenGL 环境下画出

的。而要扫描上面的过程, 采用 OpenGL 直接按扫描线颜色的画法, 对上电过程和断电过程的模拟是可以的, 但在对余辉现象进行模拟时, 由于雷达显示器有一定的底色, 在余辉由高亮度慢慢变为没有亮度时, 依靠这种直接的画法就不能够显示出我们想要的效果。所以要利用雷达显示器的底色和余辉模型生成的 HIS 采用一定的算法转化成 RGB 颜色。本文采用 OpenGL 中提供的颜色融合算法进行实现。颜色融合是将两种颜色融合在一起产生一种新的颜色效果。

OpenGL 对融合的处理是基于 RGBA 颜色模型采用的。在 OpenGL 中如果不进行颜色融合, 则每个新的颜色值被系统认为是完全不透明的 ($\alpha = 1$), 覆盖掉帧缓存中已经存在的颜色值。如果使用了融合操作, 可以控制将多少成分的既存颜色与新的颜色进行合成 (α 不等于 1), 这样就可以部分地显示先前存储的颜色。

我们把 P 型显示器分成 180 个小扇形, 从系统上电开始, 就要对这 180 个小扇形进行管理控制, 余辉现象的模拟主要是通过这些小扇形表现出来的。

扫描线是使用线条来表示的, 用户可以通过键盘控制扫描线的转动方向以及转动的快慢。

在用户使用键盘一直控制让扫描线转动时, 扫描线的位置要不停地变化, 各个小扇形的亮度也发生变化, 所以需要画 P 型显示器的函数一直运行。为了达到这一点, 我们使用一个单独的线程来画 P 型显示器。这样做的好处有两点: 一是程序在画扫描线时, 还能对用户其它的动作做出反应, 另一点是相比较于使用定时器这种方法而言的, 使用线程可以比使用定时器时间上更为精细。

当用户用鼠标点击“高压”开关, 并且“高压”开关置于开状态时, 系统开始启动 P 型显示器的绘制线程。线程中程序的具体流程如图 4 所示。

5 结束语

基于 OpenGL 三维图形标准来实现对雷达显示器的计算机模拟, 尤其采用基于 HIS 颜色模型的余辉亮

(下转第 60 页)

度模型可以很好地解决雷达显示器仿真过程中长余辉模拟的难点,从而达到理想的仿真效果。

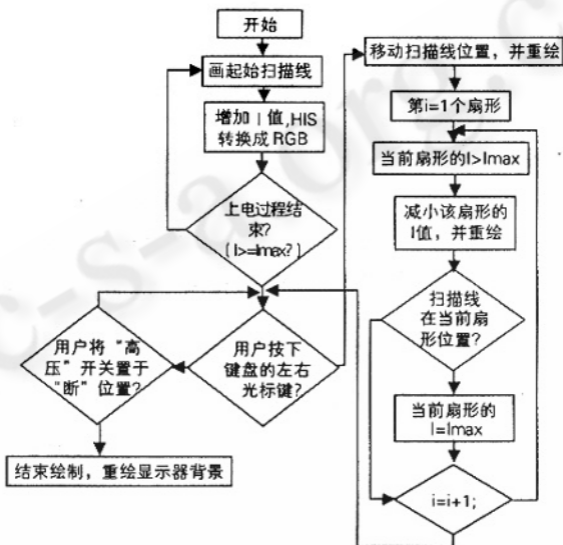


图 4

参考文献

- 1 白建军, OpenGL 三维图形设计与制作, 人民邮电出版社, 1999。
- 2 吴斌, OpenGL 编程实例与技巧, 人民邮电出版社, 1999。
- 3 陈怀瑾, 21 世纪航天与军事领域系统仿真技术发展展望, 系统仿真学报, Vol. 11 No. 5。
- 4 陈国发、冯兵, 雷达系统仿真, 雷达与对抗, 1997。
- 5 Distributed Simulation Technology Inc. GL Studio User's Guide. 2000。