

在虚拟现实快速应用中立体显示^①

吕成宾, 鲍苏苏

(华南师范大学 计算机学院, 广州 510631)

摘要: 提出了立体照相机的概念, 通过立体照相机分析了平行立体投影法的视差性质、瞳距大小及视点的视图和投影变换, 最后给出了开发立体显示的步骤。实验表明, 程序开发者只需遵循给定的步骤开发支持立体显示的虚拟现实程序, 不必经过多次的尝试便可以体验良好的立体视觉效果, 大大缩短了程序开发时间。

关键词: 立体显示; 立体照相机; 视差; 平行投影法

Rapid Application of the Stereoscopic Display in Virtual Reality

LV Cheng-Bin, BAO Su-Su

(School of Computer, South China Normal University, Guangzhou 510631, China)

Abstract: This paper proposes the concept of the stereoscopic camera. And then it is used to analyze the properties of the parallax, the distance between pupils, and the transformations of the view and projection in the parallel axis projection method. Finally this paper gives the steps of the stereoscopic development. The experiment shows that following these steps, the application developer can develop the virtual reality program supporting stereoscopic display without several attempts to obtain a good stereoscopic effect. The program development time can be greatly reduced.

Key words: stereoscopic display; stereoscopic camera; parallax; parallel axis projection

虚拟现实 (VR) 是采用以计算机技术生成逼真的视、听、触觉一体化的特定范围的虚拟环境, 用户借助必要的设备以自然的方式与虚拟环境中的对象进行交互作用、相互影响, 从而产生“沉浸”于等同真实环境的感受和体验^[1]。立体显示是 VR 的一种极重要的支撑技术, 通过此技术, 可以在 VR 中体验深度层次分明的立体视觉效果和身临其境的感觉。

产生支撑立体显示的立体图像对必须符合三维视觉习惯, 错误的设计算法产生的图像将导致垂直视差和空间变形, 以至于出现复视、视疲劳和不适。立体显示软件技术源于立体摄影的体视图投影模型。这种模型有两种变形算法: 汇聚立体投影和平行立体投影^[2]。无论是哪种投影模型, 为获得正确的图像对, 一般的方法是通过不断地修改虚拟环境中投影中心的观察与投影参数, 然后再不断地尝试。这种方法既费时又耗力, 得到的结果也可能不太理想。因此, 本文提出了立体照相机模型, 在 VR 中建立虚拟立体照

相机, 通过明确的几个步骤便能在 VR 中快速应用立体显示技术。

1 立体显示原理

人类从客观世界获得信息 80% 以上来自视觉, 其中最重要的一个视觉效果是深度的感知。但由于显示设备是二维的, 原有的三维空间具有的深度信息在经过二维投影后丢失了, 用户只能通过遮挡、纹理等心理暗示微弱地感知深度。人类对深度的感知除了心理暗示外, 还有基于双目视差的生理暗示^[3]。立体显示技术就是在计算机上利用双目视差实现立体成像。

人们通常总是用双眼同时观看物体。由于人类左右两眼有间距, 造成两眼的视角有些细微的差别, 而这样的差别会让两眼分别看到的景物有一点点的位移。大脑通过眼球的运动、调整, 综合了两眼视网膜中图像的信息, 感知到生理深度暗示, 从而产生立体感。立体显示就是根据这种原理, 先通过计算机技术

^① 收稿时间:2011-03-04;收到修改稿时间:2011-04-04

对同一空间场景产生一对具有微小差别的图像，再经过特殊的设备传输，例如双色眼镜、液晶快门眼镜、偏振眼镜、头戴式追踪装置 (Head Trackers)、头盔或者立体显示器，让左眼只观看左图像，右眼只观看右图像，从而让这个场景展现出具有立体感。左右眼观察的差异就是通常所说的双目视差，具备这种差异的两幅二维图像称为“立体图像对”。视差具有零视差、正视差和负视差之分，其立体显示效果也不一样。如图 1。当立体图像对为零视差时，两幅图像之间没任何差别，出现零视差的对象看起来位于显示设备的屏幕上。当出现正视差时，两眼观看图像对应点的视线不交叉，出现正视差的对象看上去就像位于显示设备屏幕后面的空间一样。当出现负视差时，两眼观看图像对应点的视线出现交叉，也就是视差交叉，这种情况下，被观察的物体看上去就像漂浮在眼睛和显示设备之间的区域里。

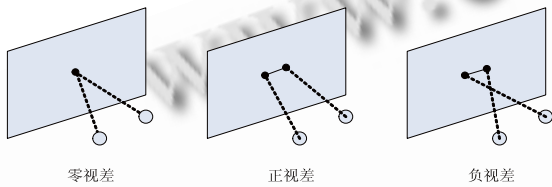


图 1 视差类型

2 立体照相机的提出

为了把一个物体的三维坐标变换为屏幕上的像素坐标，大多数图形 API 一共由以下三个步骤完成^[4]：

① 变换。包括模型、视图和投影操作。变换可使物体坐标变换为指定坐标系下的坐标，从而让后面的步骤更容易计算。

② 裁剪。由于场景是渲染在一个矩形的窗口中，位于窗口之外的物体（或部分物体）必须被裁剪掉。在三维计算机图形中，裁剪就是丢弃位于裁剪平面之外的物体。

③ 视口变换，即经过变换的坐标和屏幕上的像素之间建立对应的关系。

立体显示的目的是产生立体图像对，视差是由两个不同视点的透视投影引起的，所以上面的步骤必须执行两次，且两次执行的不同之处在于视图变换和投影变换。产生目标场景的变换过程类似于用照相机进行拍照，把场景的两个视点当作左右照相机，称为立体照相机。如图 2，立体照相机有如下属性：

① 观察属性，包括投影中心 COP (V_p)、观察方向 (V_d) 和朝上方向 (V_u)，都可以用三维空间坐标系下

的三维向量来表示。

② 瞳距 (e)，即两个照相机之间的距离。

③ 焦距 (f)，在透镜成像中，焦距是指使成像最清晰的点到光心的距离；而在这里是指在相机标架下，投影平面到 Z 轴的距离，也是 COP 到投影平面的距离。

④ 可视空间属性，相机的视景体是一个平截头体。如图 3，表示该平截头体的参数有： YZ 平面上的视野角度 (θ ，且 $0^\circ < \theta < 180^\circ$)、近裁剪面距离 (L_n)、远裁剪面距离 (L_f) 以及近裁剪面的纵横比（宽度除以高度，用 ψ 表示）。在照相机坐标系下指定近裁剪面的左下角坐标 ($L_l, L_b, -L_n$) 和右上角坐标 ($L_r, L_t, -L_n$) 也可以描述平截头体。

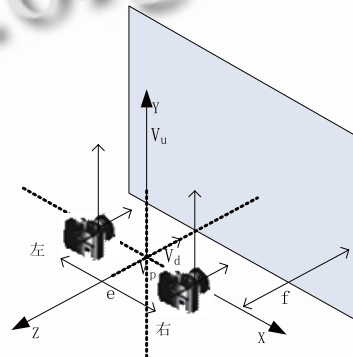


图 2 立体照相机属性

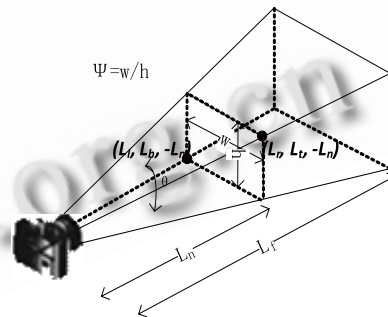


图 3 照相机的可视空间属性

3 快速应用立体显示

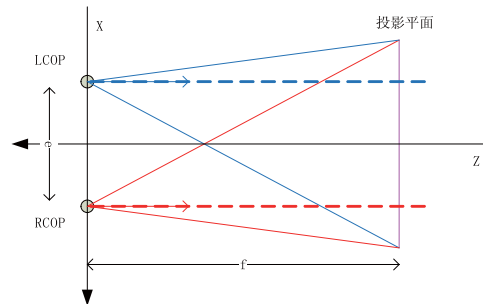


图 4 平行立体投影法

由于汇聚立体投影会产生垂直视差^[5]，所以本文主要讨论针对平行立体投影的 VR 快速应用立体显示。平行立体投影的视轴相互平行，且与投影平面垂直，如图 4。

3.1 视图与投影变换

在场景中设置好立体照相机参数后，可以分别计算左右相机的观察属性及可视空间属性，从而得出视图和投影变换。

立体照相机的视向和朝向相互垂直，并都与两个照相机的连线垂直，向量 $V = \frac{V_d \times V_u}{\|V_d \times V_u\|} \times \frac{e}{2}$ 。如图 5，可以得出左右相机的观察属性：

$$\begin{cases} V_{lp} = V_p - V \\ V_{ld} = V_d \\ V_{lu} = V_u \end{cases} \quad (1) \quad \begin{cases} V_{rp} = V_p + V \\ V_{rd} = V_d \\ V_{ru} = V_u \end{cases} \quad (2)$$

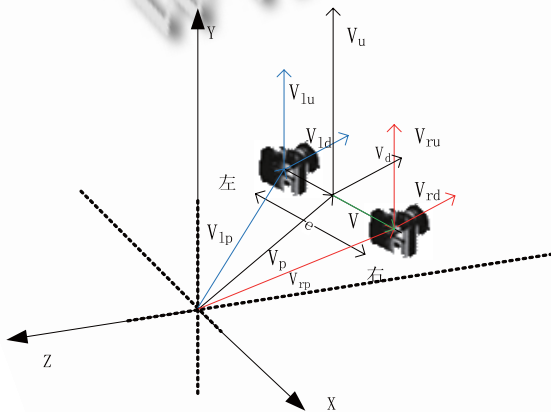


图 5 左右相机的观察属性

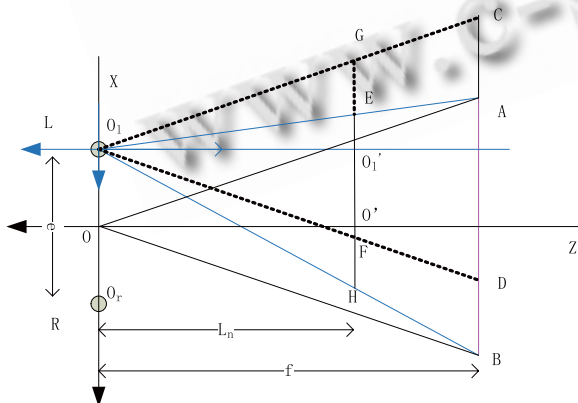


图 6 左相机的俯视图

容易看出它们的可视空间都是非对称的平截头

体。由此需要计算每个平截头体近裁剪面的左下角坐标和右上角坐标。对于左相机，如图 6，需要计算在左相机坐标系（原点在 O_l ）下的两点坐标 $E(L_{ll}, L_{lb}, -L_n)$ 和 $H(L_{lr}, L_{lb}, -L_n)$ 。作 LC 平行 OA，LD 平行 OB，则：

$$L_{AC} = L_{BD} = e/2$$

$\Delta O_l G E \sim \Delta O_l C A$ 和 $\Delta O_l F H \sim \Delta O_l D B$ ，所以有：

$$L_{GE} = L_{FH} = eL_n/2f$$

因此得出：

$$\begin{cases} L_{ll} = -L_{O_l G} + L_{GE} = -\psi L_n \tan(\theta/2) + eL_n/2f \\ L_{lr} = -L_{O_l F} + L_{FH} = \psi L_n \tan(\theta/2) + eL_n/2f \\ L_{lb} = -L_n \tan(\theta/2) \\ L_{ln} = L_n \tan(\theta/2) \end{cases} \quad (3)$$

同理，设右相机坐标系下的左下角和右上角的两点坐标为 $(L_{rl}, L_{rb}, -L_n)$ 和 $(L_{rr}, L_{rt}, -L_n)$ ，则：

$$\begin{cases} L_{rl} = -\psi L_n \tan(\theta/2) - eL_n/2f \\ L_{rr} = \psi L_n \tan(\theta/2) - eL_n/2f \\ L_{rb} = -L_n \tan(\theta/2) \\ L_{rn} = L_n \tan(\theta/2) \end{cases} \quad (4)$$

3.2 视差大小及性质

视差的大小实际上三维坐标经过两个视点变换为屏幕上的像素坐标的距离。如图 7，对于在标准标架下的点 $P(x, y, z)$ ，先变换到左相机标架下的点 $P_l(x_l, y_l, z_l)$ ；接着根据透视成像原理，投影到左投影平面上的点为 $P_l'(x_l', y_l')$ ；最后经过视口变换后在屏幕上的点为 $P_l''(x_l'', y_l'')$ 。则以下公式成立：

$$\begin{cases} x_l = x + e/2 \\ y_l = y \\ z_l = z \end{cases} \quad (5) \quad \begin{cases} x_l' = -x_l f / z_l = -(x + e/2)f / z \\ y_l' = -y_l f / z_l = -yf / z \end{cases} \quad (6)$$

$$\begin{cases} x_l'' = x_l' - e/2 = -(x + e/2)f / z - e/2 \\ y_l'' = y_l' = -yf / z \end{cases} \quad (7)$$

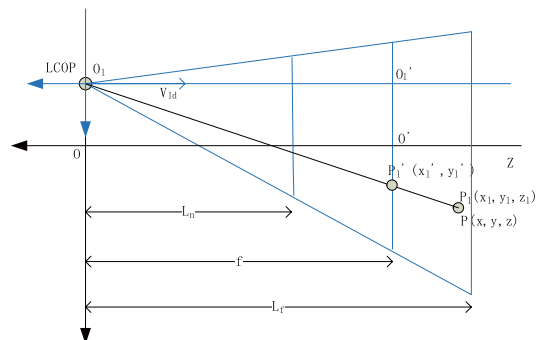


图 7 三维空间点 P 在左相机标架下的投影变换

同理，可以得出点 P 经过右相机后的一系列变换和视口变换后，在屏幕上生成的点 $P_r''(x_r'', y_r'')$ ，则

$$\begin{cases} x_r'' = -(x - e/2)f/z + e/2 \\ y_r'' = -yf/z \end{cases} \quad (8)$$

由公式(7)、(8)得出水平和垂直视差 P_h 、 P_v (其中 $z < 0$):

$$\begin{cases} P_h = x_r'' - x_r' = e(1 + f/z) \\ P_v = y_r'' - y_r' = 0 \end{cases} \quad (9)$$

由此可以得出以下性质:

- ① 平行立体投影法不会产生垂直视差。
- ② 视差的大小与 e 、 L_n 、 f 、 z 相关。立体照相机标架下的三维场景中的任意点，其 z 值越小视差越大。

③ 当 $z = -f$ 时， $P_h = 0$ ； $z < -f$ 时， $P_h > 0$ ； $z > -f$ 时， $P_h < 0$ 。即，投影平面为零视差平面，点位于该平面则立体效果是在屏幕上。点在零视差平面和近裁剪面之间则将会出现负视差，即点的立体效果是在屏幕和观察者空间上。点在零视差平面和远裁剪面之间则将会出现正视差，即点的立体效果是在屏幕后面。

3.3 瞳距大小

由于支持立体显示的设备多样化，有小屏幕也有大屏幕，所以对于 P_h 不适合用距离的大小来描述，使用水平可视化角度 (HVA) 进行描述更为合适，如图 8。

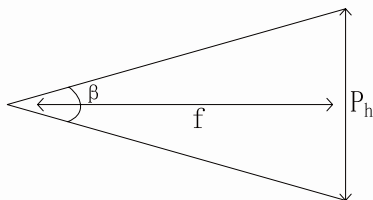


图 8 视差的水平可视化角度

由此，得：

$$P_h = 2f \tan(\beta/2) \quad (10)$$

由公式 9 得(其中 $z < 0$ ， $i = z/f$):

$$e = P_h \times \frac{z}{z+f} = P_h \times \frac{z/f}{1+z/f} = 2f \tan(\beta/2) \times \frac{i}{1+i} \quad (11)$$

在立体显示时，虽然感兴趣点的深度不同于屏幕，仍要求我们聚焦于显示屏。在这种情况下，Valyus 发现大多数人能容忍的汇聚变化角度为 1.6° ，如果增大角度，会引起“调节/汇聚”矛盾。Hodges 在结合 Valyus 和 Yeh 在时分立体显示研究的基础上，为观察者融合两张视差图像而且尽量减少双目干扰 (cross talk) 现

象，提出了 HVA 的最大值为 1.5° ^[6]。由此从公式(11)中，可以得出最大瞳距：

$$e_{\max} = 2f \tan(1.5^\circ/2) = 0.028f \quad (12)$$

根据上面的推论，在虚拟现实快速应用立体显示技术的步骤如下：

- ① 选择立体相机的视野角度，一般是 $45^\circ - 60^\circ$ 之间。
- ② 选择立体相机的近裁剪面、远裁剪面和焦距长度，即零视差位置，使得场景中的近对象位于屏幕之外，而远对象位于屏幕之内，最近的对象不能超过焦距的一半。选择瞳距距离，使其不超过焦距的 0.028 倍。根据窗口大小设置相机纵横比。
- ③ 在场景中摆放立体相机，包括位置、视向、朝向。
- ④ 利用公式 1~4 计算 LCOP 的视图和投影变换，然后进行对象建模，最终光栅化之后在缓冲区中存储左视图。
- ⑤ 以同样的方式处理 RCOP，在缓冲区中存储右视图。
- ⑥ 左右视图经过立体设备最终实现立体显示。视觉结果的调节只需可对 f 或 e 进行微调，然后再重复第五、六步。

4 实验结果及分析

本文通过图形工作站 (CPU 为 AMD 2.60GHz*4; 内存为 4GB; 显卡为 FX4600) 与 NVIDIA 3D 立体幻境和三星 2233RZ 120Hz 显示器构建基于时分立体显示系统。图形 API 使用 OpenGL。在实验中，设计立体照相机的参数如表 1:

表 1 立体照相机参数

参数	值	参数	值
θ	50°	V_p	(0,0,3.144)
L_n	2.144	V_d	(0,0,-3.144)
L_f	4.144	V_u	(0,1,0)
f	3.144	ψ	1.0
e	0.088		

立体照相机位于(0,0,3.144)，指向 Z 轴负方向，且焦距为 3.144。所以， $z=0$ 的对象出现在零视差平面上； $z > 0$ 的对象出现负视差； $z < 0$ 的对象出现正视差。实验的过程是在三维场景中放置五个对象，如表 2。实验结果如图 9、10，实验表明，5 个对象立体层次分明，且不会产生复视，具有较好的视觉效果。

表2 5个对象在场景中的位置

对象	位置 (x,y,z)	效果
立方体	-0.5,0.5, 0.8	负视差
黄色球体	0.5,0.5, 0.5	负视差
圆锥体	0.0,0.0,0.0	零视差
红色球体	-0.8,-0.8, -0.5	正视差
茶壶	0.8,-0.8, -0.8	正视差

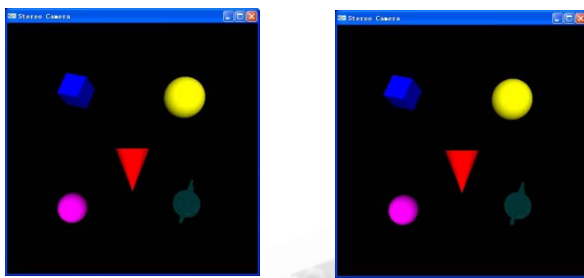


图9 左图是生成的左视图, 右图是生成的右视图

5 结语

立体显示技术使产生真正的具有立体感的图形成为可能, 其良好的三维效果引起了人们的普遍重视,

具有广阔的应用前景。近年来, 3D 立体技术在 3D 电影和微机平台上有很大的发展, 特别是在 NVIDIA 3D 立体幻境的推出后, 便有一百多款的 3D 游戏支持立体显示。本文提出的立体照相机模型能帮助开发者快速应用立体显示技术, 具有重要意义。

参考文献

- 1 洪炳镕,蔡则苏,唐好选.虚拟现实及其应用.北京:国防工业出版社,2005.1-10.
- 2 顾郁莲,蔡宣平.计算机立体视图绘制技术.国防科技参考, 1998,19(1):63-70.
- 3 支丽欧,立体图像对的生成算法研究[硕士学位论文].重庆:重庆大学,2007.
- 4 Shreiner D. OpenGL 编程指南.第5版.北京:机械工业出版社,2006.66-68.
- 5 杨小草.微机立体动画研究及体视误差分析.计算机辅助工程,1996(1):31-39.
- 6 Hodges LF. Tutorial: Time-Multiplexed Stereoscopic Computer Graphics. IEEE Computer Graphics and Applications, 1992,12(2):20-30.
- (上接第 179 页)
- Image Enhancement. Graphical Models and Image Processing, 1996,(58):82-89.
- 7 Paul K, Choudhury DR, Chaudhuri PP. Cellular Automata Based Transform Coding for Image Compression, HiPC99, LNCS 1745, 1999. 269-273.
- 8 王宏,张运杰,魏华.基于元胞自动机模型的图像分割算法,模糊系统与数学,2004,18:309-313.
- 9 Zhang K, Li Z. Edge Detection of Images based on Fuzzy Cellular Automata. International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, Qing Dao, Aug. 2007,2.
- 10 Yu YC, Wang JD, Ding J. An Extended Model of the Cellular Automata, Computing, Communication, Control, and Management, 2009. CCCM. 2009. 66-69.
- 11 Munshi S. A Cellular Automata based Network Security Model. 2011 Emerging Applications of Information Technology, 2011. 275-279.
- 12 丁永生,邵世煌,万庆萱.采用元胞自动机机理的针织电脑编织系统.中国纺织大学学报,1997,23(1):6-11.
- 13 张芳,丁永生,耿兆丰.基于模糊元胞自动机的复杂图案、纹理生成的实现.中国纺织大学学报,1999,25(5):32-34.
- 14 Wang H, Jiang JN, Zhou LH. Image Processing Based on Fuzzy Cellular Automata Model. 2009 Fourth International Conference on Innovative Computing, Information and Control. ICICIC. 2009. 954-957.
- 15 Zhang K, Yuan JS, Guo J. Production of Patterns Based on Fractal Fuzzy Cellular Automata. Proc. of the 27th Chinese Control Conference, Kunming, Yunnan, China, July, 16-18, 2008.245-248.
- 16 王仲君,王能超,等.元胞自动机的演化行为研究.计算机应用研究,2007,24(8):38-41.