

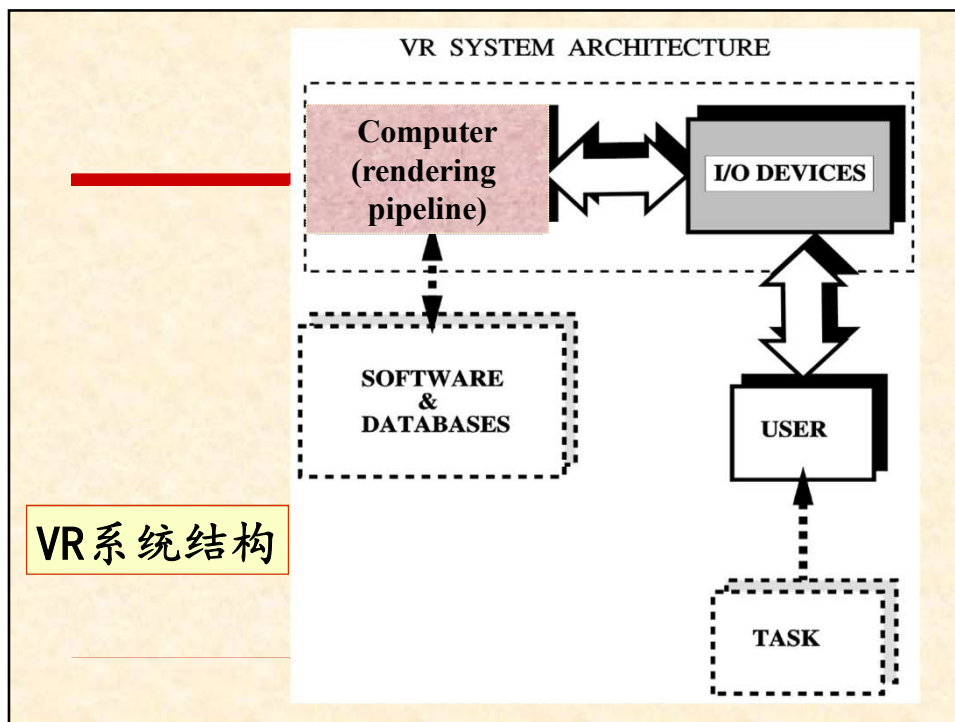
虚拟现实技术

第四章 虚拟现实系统的体系结构

(1) VR绘制流水线

本章主要内容

- ◆ 虚拟现实引擎
 - ◆ 图形绘制流水线
 - ◆ 绘制流水线应用阶段
 - ◆ 绘制流水线几何阶段
 - ◆ 绘制流水线光栅阶段
 - ◆ 流水线的平衡
-



虚拟现实引擎

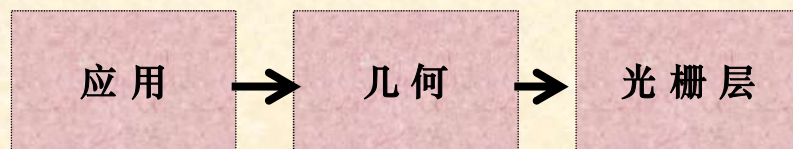
- ◆ 定义：**虚拟现实引擎**是任何VR系统的关键部分，它从输入设备中读取数据，访问与任务相关的数据库，执行任务要求的实时计算，从而实时更新虚拟世界的状态，并把结果反馈给输出显示设备。

虚拟现实引擎要求

- ◆ 快速的图形及触觉刷新率（图形30帧/秒，对于触觉几百Hz）；
- ◆ 低延迟性（<100 ms 防止病态仿真）
- ◆ VR引擎中，核心是绘制流水线
- ◆ 绘制包括触觉

图形绘制流水线

- ◆ 从3-D模型产生2-D场景的过程，称为**绘制**（rendering）
- ◆ 绘制流水线有三个功能性阶段. 流水线的速度由最慢阶段决定.



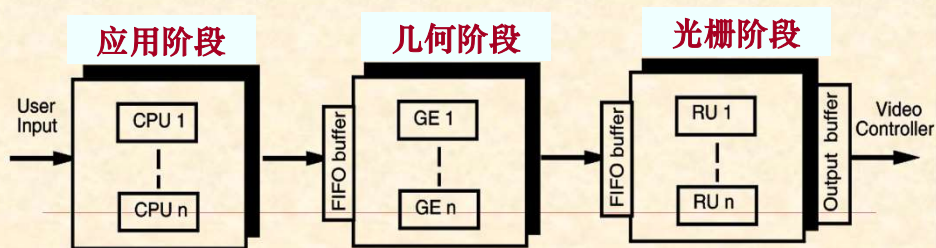
绘制流水线的三个功能阶段

图形绘制流水线

◆ 旧绘制流水线由软件完成（速度很慢）

◆ 现代的流水线结构使用平行和缓冲

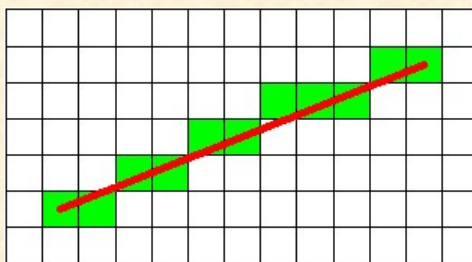
- ✔ 应用层阶段用软件完成
- ✔ 其它阶段用硬件加速完成



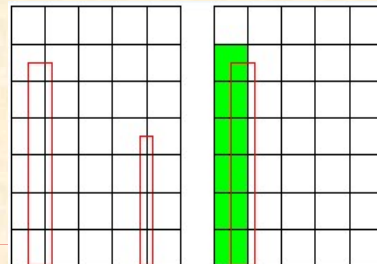
走样现象

◆ 在光栅图形显示器中，对于非水平且非垂直的直线或多边形边界进行扫描转换时，会呈现锯齿状，如下图所示，这种用离散量表示连续量引起的失真，就叫做走样 (Liasing)

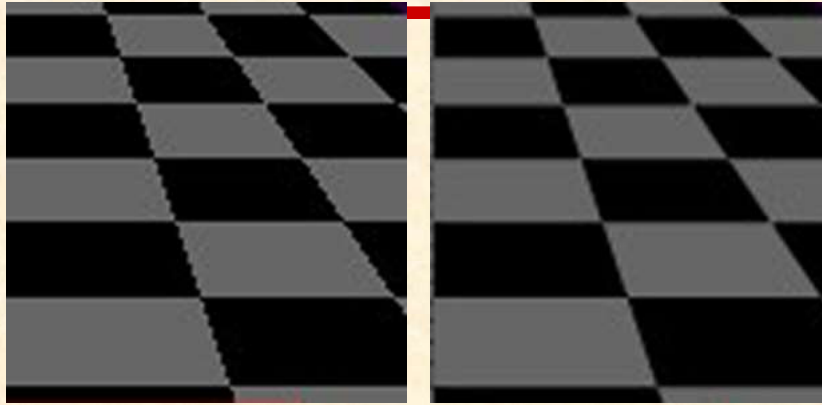
锯齿现象



丢失细节



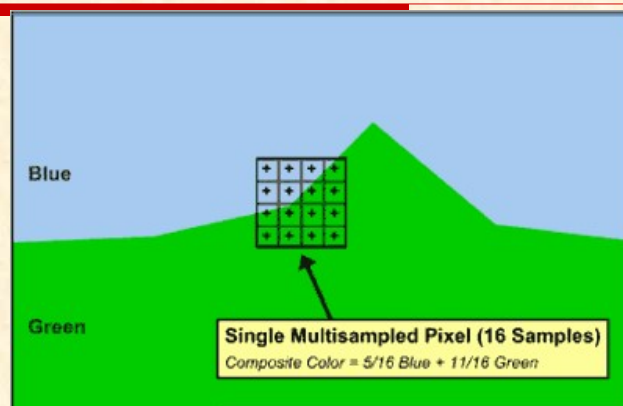
现代的流水线也做了对点、线及场景的反走样处理



走样多边形
(锯齿边)

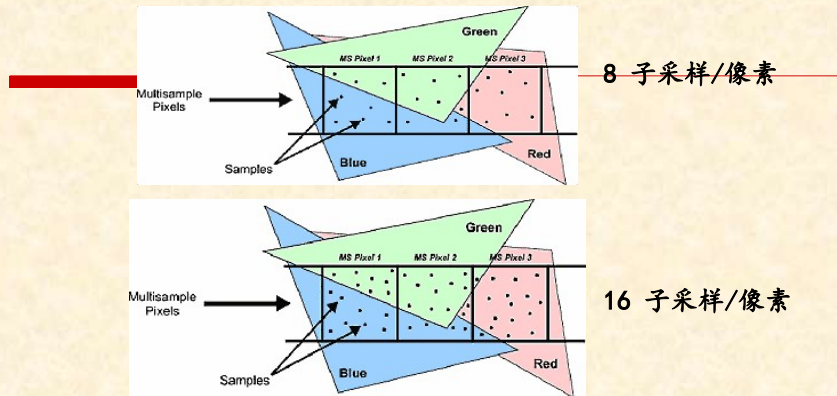
反走样多边形

- 反走样怎样完成的？ 每个象素被进行子分割
- (子采样) 成n快，每个子像素具有一种颜色；



该反走样像素是绿蓝色 (5/16 blue + 11/16 green).
无子采样像素完全绿色 - 像素中心的颜色

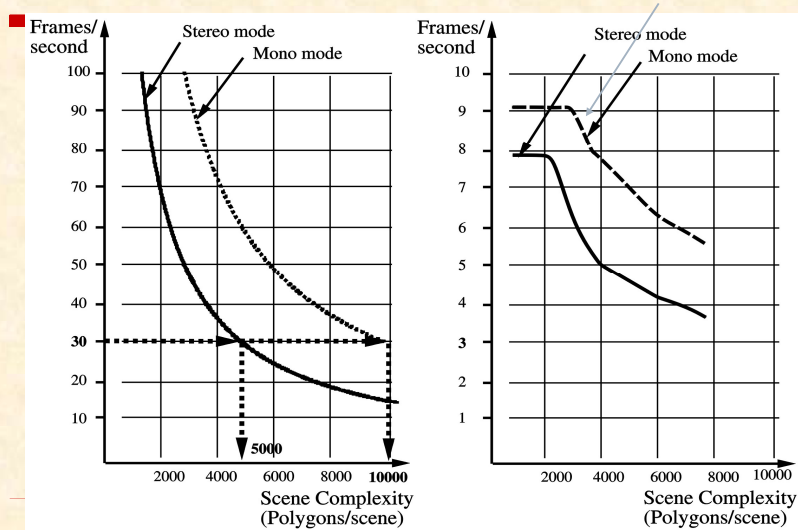
■ 多采样产生较好的反走样效果



Multisample Pixel	Actual Color Makeup			16 Sample Final Color			8 Sample Final Color		
	Red	Green	Blue	Red	Green	Blue	Red	Green	Blue
MS Pixel 1	0%	47%	53%	0%	50%	50%	0%	37%	62%
MS Pixel 2	17%	59%	24%	12%	62%	25%	12%	62%	25%
MS Pixel 3	86%	10%	4%	87%	6%	6%	100%	0%	0%

理想流水线的输出与真实流水线的输出关于场景复杂度函数的对比
(流水线瓶颈的影响)

HP 9000 workstation



绘制流水线应用阶段

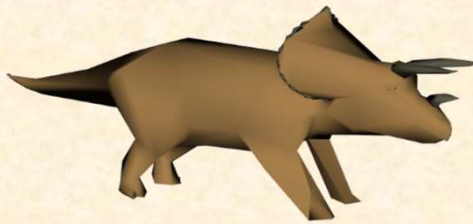


应用阶段

- ◆ 全部由CPU用软件完成
 - ◆ 读输入设备（例如数据手套、鼠标）；
 - ◆ 改变虚拟摄像机的坐标；
 - ◆ 对触觉设备执行碰撞检测并对碰撞进行反应
（基于对象的属性）
 - ◆ 力反馈作为碰撞反应的一种
-

应用阶段的优化

- ◆ 降低模型的复杂度（具有较少多边形的模型）



低分辨率模型
约 600 个多边形

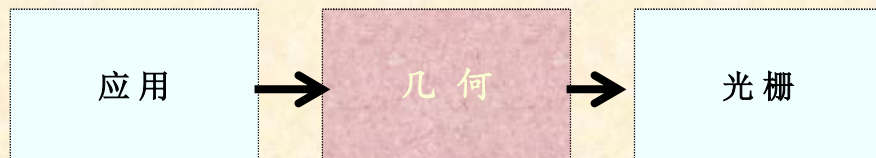


高分辨率模型
约134,754 个多边形

应用阶段优化（续）

- ◆ 减少浮点运算精度(单精度而不用双精度)
- ◆ 降低除法运算次数
- ◆ 因为由CPU完成，应该增加双处理器的速度(超规模的结构)

绘制流水线几何阶段



几何阶段

◆ 由硬件完成；

◆ 步骤

- ✓ 建模和视图变换构成；
- ✓ 在光照模型的基础上建立阴影；
- ✓ 最后场景被投影、裁剪并映射到屏幕坐标

光照子阶段

◆ 第一步，根据下列因素计算表面颜色

- ✔ 模拟光源的种类和数量
- ✔ 照明模型
- ✔ 反射表面属性
- ✔ 大气的效果，如雾或烟

◆ 第二步，在对象遮挡情况下的光照结果，增强真实感

什么是光照模型

◆ 光照模型(Illumination Model):计算某一点的光强度的模型

真实感图形的特点

- ◆ 能反映物体表面颜色和亮度的细微变化
 - ◆ 能表现物体表面的质感
 - ◆ 能通过光照下的物体阴影，极大地改善场景的深度感和层次感，充分体现物体间的相互遮挡关系
 - ◆ 能模拟透明物体的透明效果和镜面物体的镜面效果
-

光源

- ◆ 几何性质
 - ✔ 点光源
 - ✔ 线光源
 - ✔ 面光源
 - ◆ 光谱组成
 - ✔ 白色光等能量的各种波长可见光的组合
 - ✔ 彩色光
 - ✔ 单色光
-

简单光照明模型

- 简单光照明模型亦称局部光照明模型，其假定物体是不透明的，只考虑光源的直接照射，而将光在物体之间的传播效果笼统地模拟为环境光

简单光照明模型

光照射到物体表面，主要发生：

反射

透射（对透明物体）

部分被吸收成热能

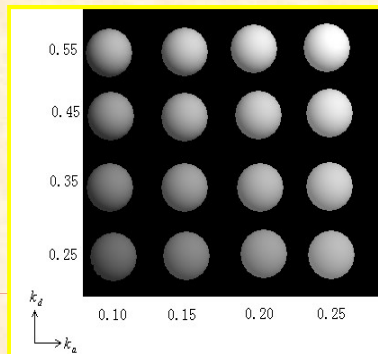
反射光，透射光决定了物体所呈现的颜色

简单光照模型-漫反射

- ◆ 将环境光与漫反射结合起来

$$I = I_e + I_d = I_a K_a + I_p K_d (L \cdot N)$$

一般取 $I_a = (0.02 \sim 0.2) I_d$



$$I_a K_a$$

- ◆ **环境光**：在空间中近似均匀分布，即在任何位置、任何方向上强度一样，记为 I_a
- ◆ **环境光反射系数 K_a** ：在分布均匀的环境光照射下，不同物体表面所呈现的亮度未必相同，因为它们的环境光反射系数不同

$I_p K_d (L \cdot N)$ 漫反射-角度余弦的推导

◆ 漫反射

✓ 粗糙、无光泽物体（如粉笔）表面对光的反射

✓ 光照明方程

$$I_d = I_p K_d \cos \theta \quad \theta \in [0, \frac{\pi}{2}]$$

◆ I_d 漫反射的亮度

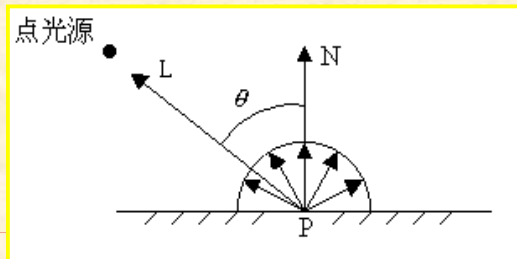
◆ I_p 点光源的亮度

◆ K_d 漫反射系数

◆ θ 入射角

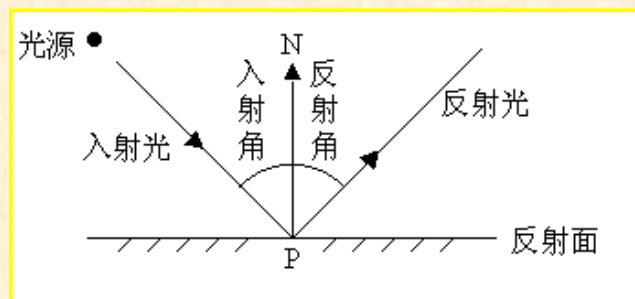
漫反射光的强度

只与入射角有关



简单光照明模型-镜面反射

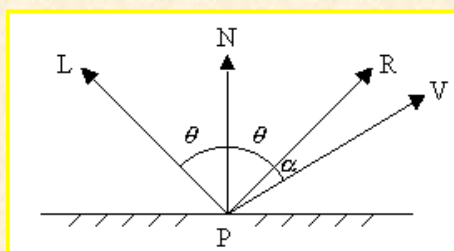
◆ 理想镜面反射



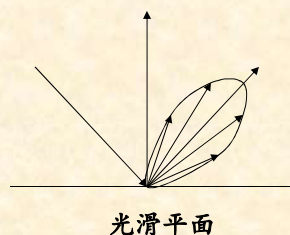
◆ 观察者只能在反射方向上才能看到反射光，偏离了该方向则看不到任何光。

简单光照模型-镜面反射

◆ 非理想镜面反射



$$I = I_p K_s \cos^n \alpha$$



- ◆ P为物体表面上一点，L为从P指向光源的单位矢量，N为单位法矢量，R为反射单位矢量，V为从P指向视点的单位矢量

简单光照模型-镜面反射

$$I_s = I_p K_s \cos^n \alpha \quad \text{或} \quad I_s = I_p K_s (V \cdot R)^n$$

- ◆ I_s 为镜面反射光强， I_p 为点光源的亮度
- ✓ K_s 是与物体有关的镜面反射系数。 n 为镜面反射指数， n 越大，则 I_s 随 α 的增大衰减的越快。
 - ✓ n 的取值与表面粗糙程度有关。
 - n 越大，表面越平滑（散射现象少，稍一偏离，明暗亮度急剧下降）
 - n 越小，表面越毛糙（散射现象严重）

简单光照模型-Phong光照模型

- ◆ Phong光照模型的综合表述：由物体表面上一点 P 反射到视点的光强 I 为环境光的反射光强 I_e 、理想漫反射光强 I_d 和镜面反射光 I_s 的总和。

$$\begin{aligned} I &= I_e + I_d + I_s \\ &= I_a K_a + I_p [K_d (L \cdot N) + K_s (V \cdot R)^n] \end{aligned}$$

简单光照模型-彩色场景的产生

◆ 产生彩色

- ✓ 选择合适的颜色模型——RGB模型
- ✓ 为颜色模型中的每一种基色建立光照方程

$$\begin{cases} I_R = I_{aR} K_{aR} + f(d) I_{pR} [K_{dR} (L \cdot N) + K_{sR} (V \cdot R)^n] \\ I_G = I_{aG} K_{aG} + f(d) I_{pG} [K_{dG} (L \cdot N) + K_{sG} (V \cdot R)^n] \\ I_B = I_{aB} K_{aB} + f(d) I_{pB} [K_{dB} (L \cdot N) + K_{sB} (V \cdot R)^n] \end{cases}$$

简单光照明模型-多个光源

◆ 采用多个光源

✓ 采用m个光源的光照明方程

$$I_{\lambda} = K_a C_{d\lambda} I_{a\lambda} + \sum_{i=1}^m f(d_i) I_{p_i\lambda} [K_d C_{d\lambda} (L_i \cdot N) + K_s C_{s\lambda} (V \cdot R_i)^n]$$

Phong光照明模型的不足

◆ Phong光照明模型是真实感图形学中提出的第一个有影响的光照明模型

◆ 经验模型，Phong模型存在不足：

- ✓ 显示出的物体象塑料，无质感变化
- ✓ 没有考虑物体间相互反射光
- ✓ 镜面反射颜色与材质无关
- ✓ 镜面反射大入射角失真现象

Gouraud着色方法

- ◆ Gouraud于1971年提出，又被称Gouraud明暗处理
 - ◆ 基本思想：在每个多边形顶点处计算颜色，然后在各个多边形内部进行线性插值，得到多边形内部各点颜色。即它是一种颜色插值着色方法。
 - ◆ 注意：Gouraud着色方法并不是孤立的处理单个多边形，而是将构成一个物体表面的所有多边形（多边形网格）作为一个整体来处理。
-

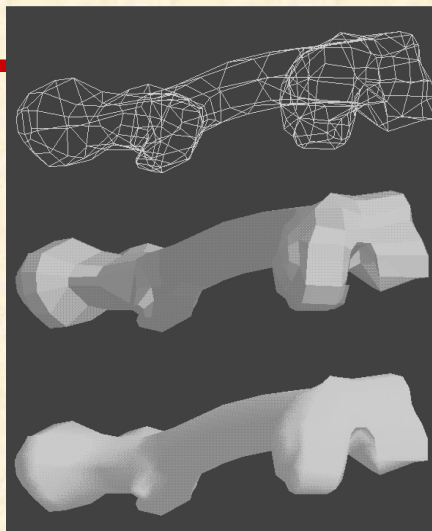
光照子阶段的优化

- ◆ 尽量进行较少的计算，减少场景的光源；
 - ◆ 阴影模型越简单，计算越少（真实感降低）：
 - ✔ 线框模型；
 - ✔ 平面着色模型；
 - ✔ Gouraud 阴影；
 - ✔ Phong 阴影。
-

光照计算模型

- ◆ **线框**是最简单的方法，它只显示多边形可见边，速度快；
- ◆ **平面着色模型**把物体的一个多边形（或面）内的所有像素赋予相同的颜色；
- ◆ **Gouraud或平滑阴影模型**在多边形内部基于边上的颜色进行插值；
- ◆ **Phong阴影模型**在计算前，对定点的法向量进行插值，然后，基于已述的光照模型进行计算—更加真实的光照模型。

光照计算模型实例



线框模型

Wire-frame model

平面着色模型

Flat shading model

Gouraud阴影模型

Gouraud shading model

绘制速度与多边形类型相关

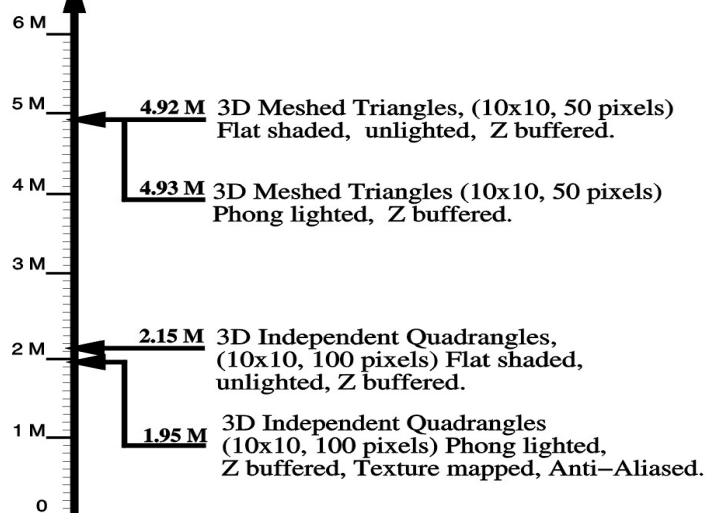
◆ 表面的描述形式影响着绘制的速度

- ✓ 如果表面用三角形网格描述，绘制比用四边形或其它多边形要快，这主要由于图形卡对于绘制三角形性能最好
- ✓ 图形卡结构是按照三角形进行优化

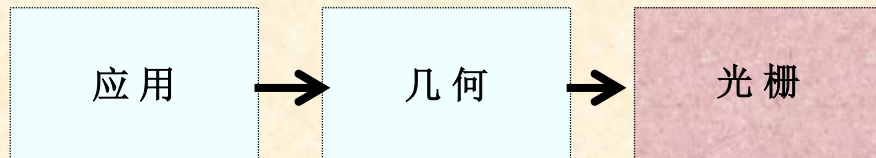
◆ 例如 SGI 绘制引擎

SGI Onyx 2

Rendering speed
(Million/sec.)



绘制流水线光栅阶段



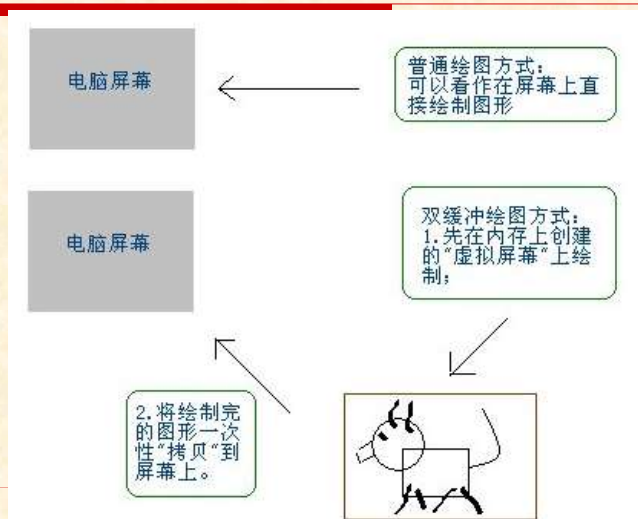
光栅阶段

- ◆ 为了速度快用硬件完成
- ◆ 把几何阶段得到的信息(3D坐标 x, y, z , 颜色, 纹理)转换为屏幕信息;
- ◆ 像素的颜色信息在颜色缓冲区中
- ◆ 像素的Z坐标存于Z-缓冲区中(与颜色缓冲区大小相同)
- ◆ 在摄像机所在的视点处可见的图元被绘制

光栅阶段（续）

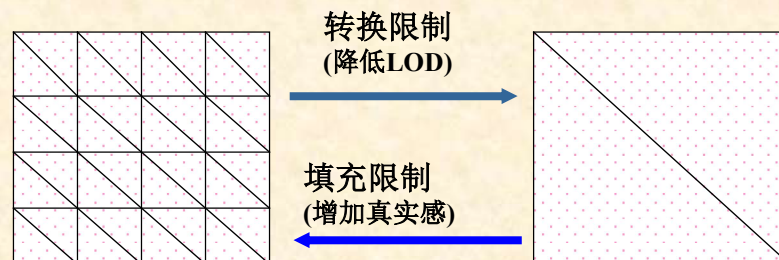
- ◆ 场景被绘制在后缓冲中
- ◆ 然后与存储将绘制图像的前缓冲区进行交换
- ◆ 这样可以消除闪烁，成为“双缓冲”
- ◆ 所有的系统缓冲分成不同帧缓冲组

双缓冲示意图



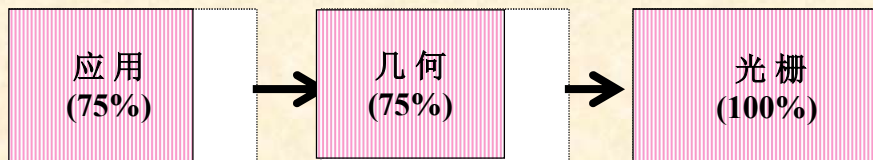
对于流水线瓶颈的测试

- ◆ 如果CPU以100%运转-流水线为“CPU限制”（瓶颈出现在应用阶段）
- ◆ 当所有光源被移除时，绘制性能能够提高，那么流水线为“转换限制，（瓶颈出现在几何阶段）
- ◆ 如果显示器的分辨率或显示规格减少时，绘制性能能够可以提高，那么流水线为“填充限制”，（瓶颈出现在光栅阶段）



流水线的平衡

单缓冲



双缓冲, 平衡的流水线



触觉绘制流水线

- ◆ 计算力和机械纹理的过程与触觉反馈有关
- ◆ 用硬件和软件完成，也有三个阶段

