



逻辑教育  
Logic education

# Hello CC

OpenGL ES 主题[3]

## 视觉班—OpenGL ES GLSL 加载几何图形

课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护



## 基本图形硬件流水线设计



### 应用程序层 -> 硬件抽象层 -> 硬件层

**应用层：**游戏和应用层软件开发人员为主体，通过调用**API**进行上层开发，不需要考虑移植性问题。

**硬件抽象层：**抽象出硬件的加速功能，进行有利于应用层开发的封装，并向应用层开发**API**

**硬件层：**将硬件驱动提供给抽象层，以实现抽象层加速功能的有效性。



**Shader**，中文名，着色器。着色器其实就是一段在GPU运行的程序。我们平时的程序，是在CPU运行。由于GPU的硬件设计结构与CPU有着很大的不同，所以GPU需要一些新的编程语言

渲染流水线分为两种，其中一种为**可编程渲染流水线**。另外一种为**固定渲染流水线**。（也称可编程管线或固定管线，管线就是流水线的意思）。渲染流水线可否编程，取决于程序猿能否在定点着色器以及片段着色器上进行编码。而现在的渲染流水线，基本都是可编程的，当然，它们也支持固定渲染流水线的功能



## 渲染流水线的具体流程

### 1、应用层

应用程序层主要与内存，CPU打交道，诸如碰撞检测，场景图监理，视锥裁剪等经典算法在此阶段执行。在阶段的末端，几何体的数据（顶点坐标，法向量，纹理坐标，纹理）等通过数据总线传送到图形硬件

### 2、硬件抽象层

在这一层，我们目前使用的是DirectX与OpenGL。对于这一部分，主要是一些API等的调用

### 3、硬件层

硬件层在渲染流水线中最为复杂，也最为重要。可编程渲染流水线与固定渲染流水线的区别在于是否对着色器进行编程。

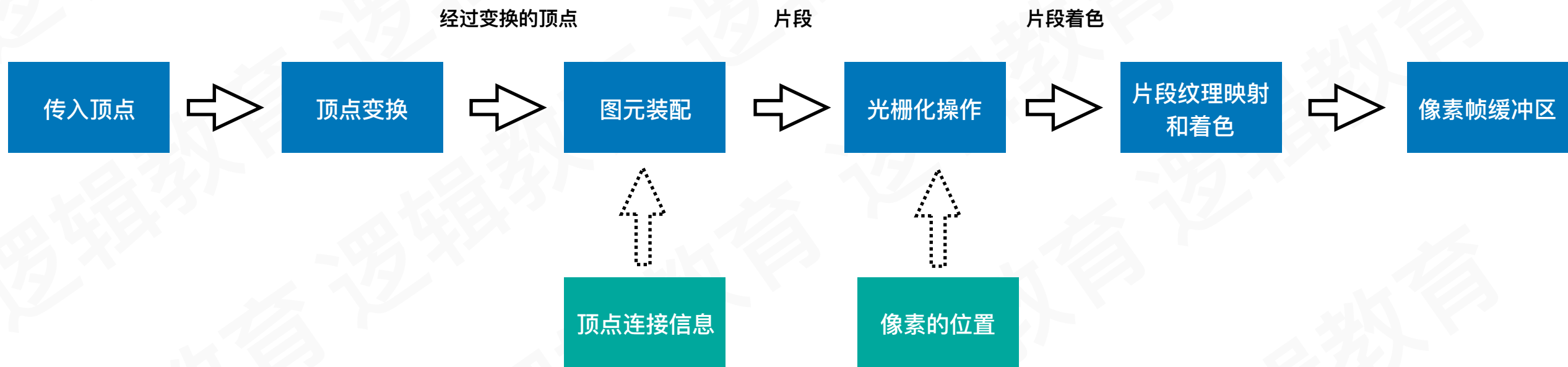
首先我们先了解固定渲染流水线它主要分为以下几个阶段：顶点变换->图元转配与光栅化->片段纹理映射和着色->光栅化操作

课程研发:CC老师

课程授课:CC老师

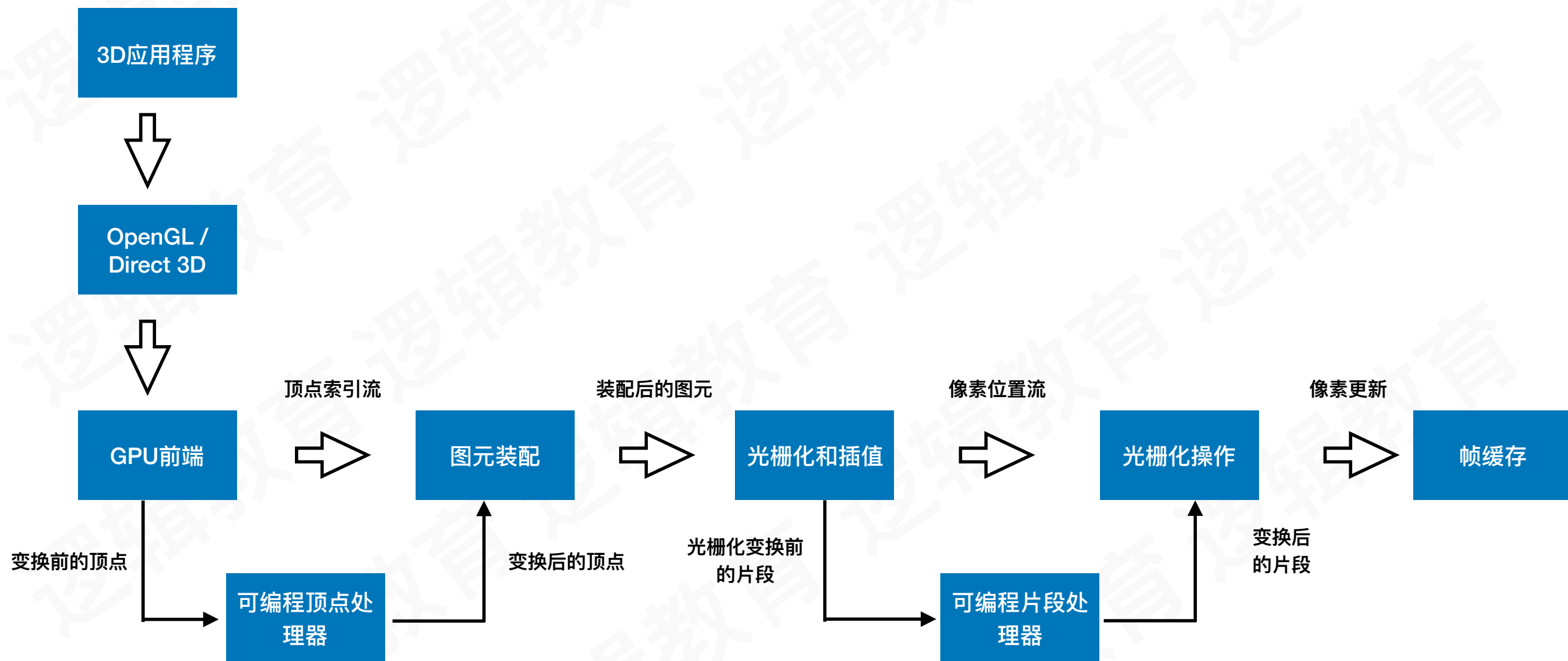


## 硬件层—固定渲染管线流程图





## 硬件层—可编程渲染流水线流程图



课程研发:CC老师  
课程授课:CC老师



逻辑教育  
Logic education

## GPU图形渲染管线

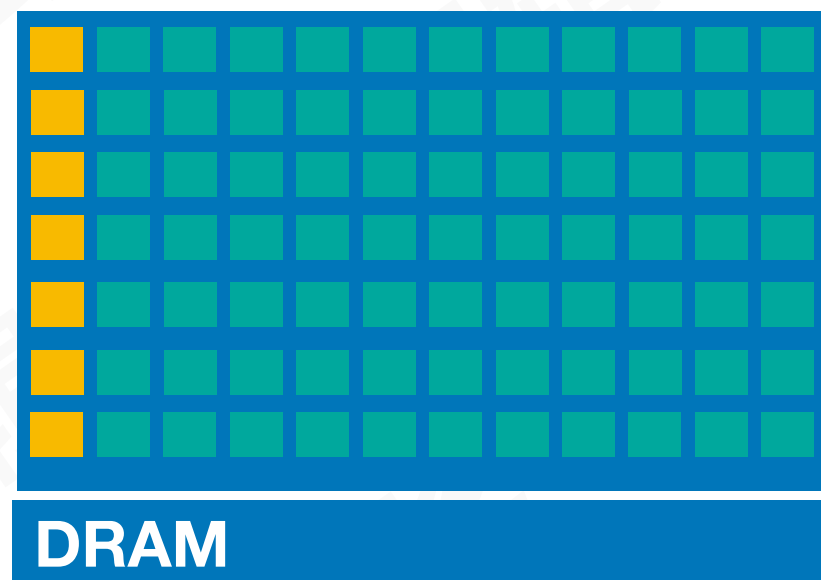


课程研发:CC老师  
课程授课:CC老师

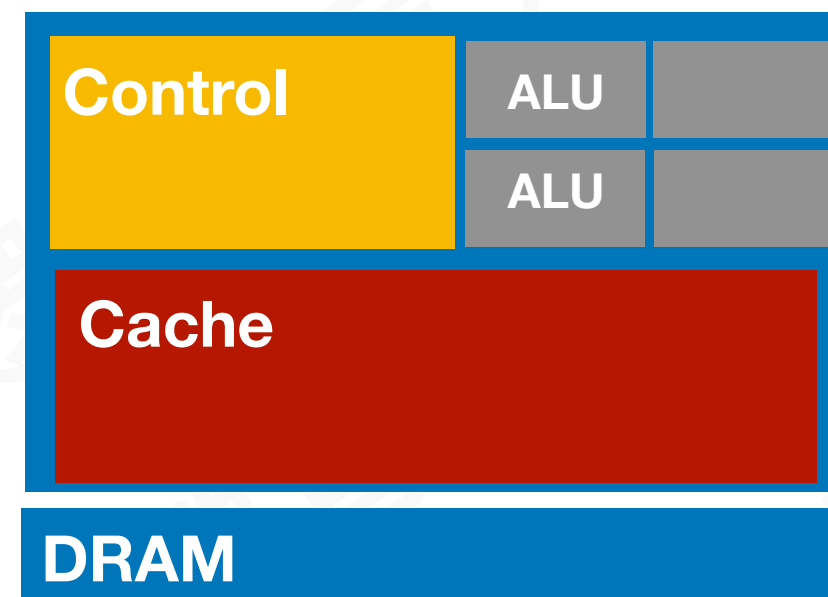
转载需注明出处,不得用于商业用途.已申请版权保护

## CPU 与 GPU 之间的关系

GPU具有高并行的结构，所以在处理图形数据和复杂算法比CPU更加有效率



GPU



CPU

CPU在执行任务的时候，一个时刻只会处理一个数据，不存在真正意义上的并行，而GPU则有多个处理器核，在一个时刻可以并行处理多个数据

课程研发:CC老师  
课程授课:CC老师





逻辑教育  
Logic education

思考

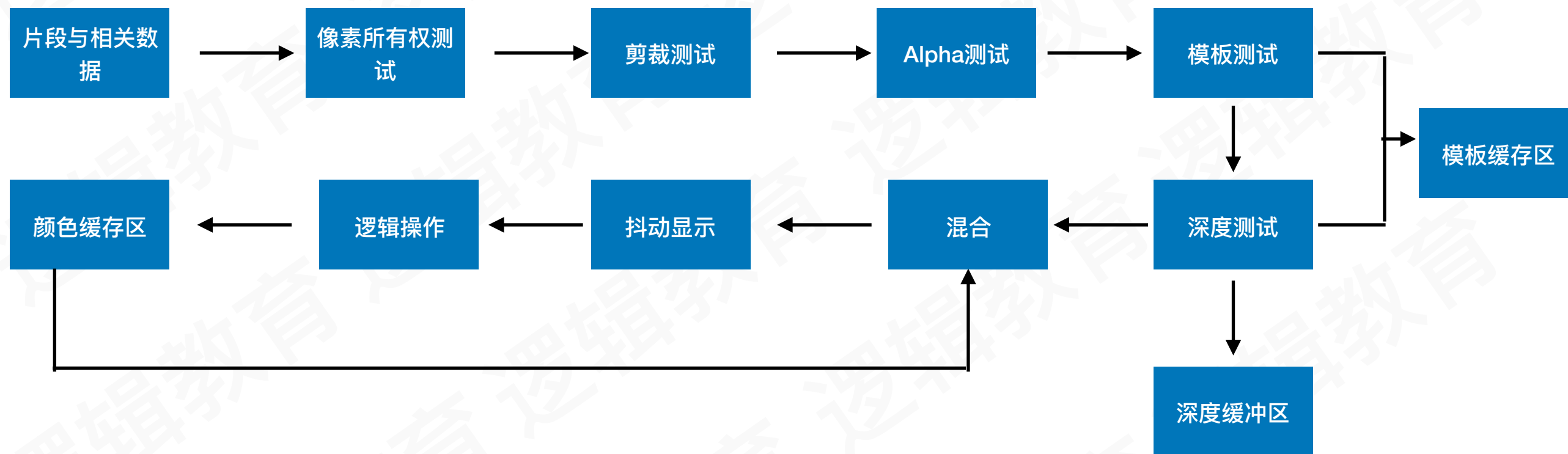
**GPU能替代CPU吗?**

课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护

## 逐片段操作

这一步我们将会对其进行各种测试，而假如它通过了所有的测试，片段将会显示在屏幕上



抖动显示：一种能够使用较少的颜色种类模拟较多颜色的显示模式

课程研发:CC老师  
课程授课:CC老师



## 几何处理阶段

几何阶段主要负责大部分多边形操作和顶点操作，包括顶点着色、坐标变换、生成图元、投影、裁剪、屏幕映射等过程，其中顶点着色、坐标变换由顶点着色器完成

### 1.顶点着色器

### 2.生成图元

### 3.投影

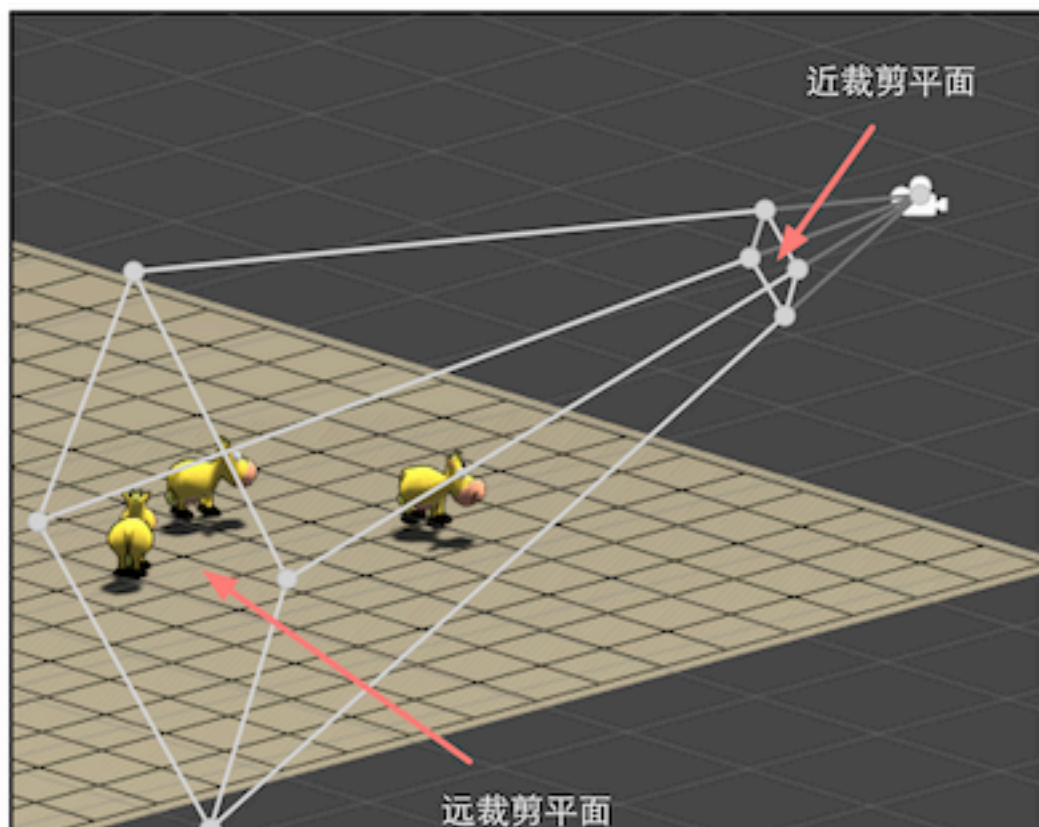
### 4.裁切

### 5.屏幕映射

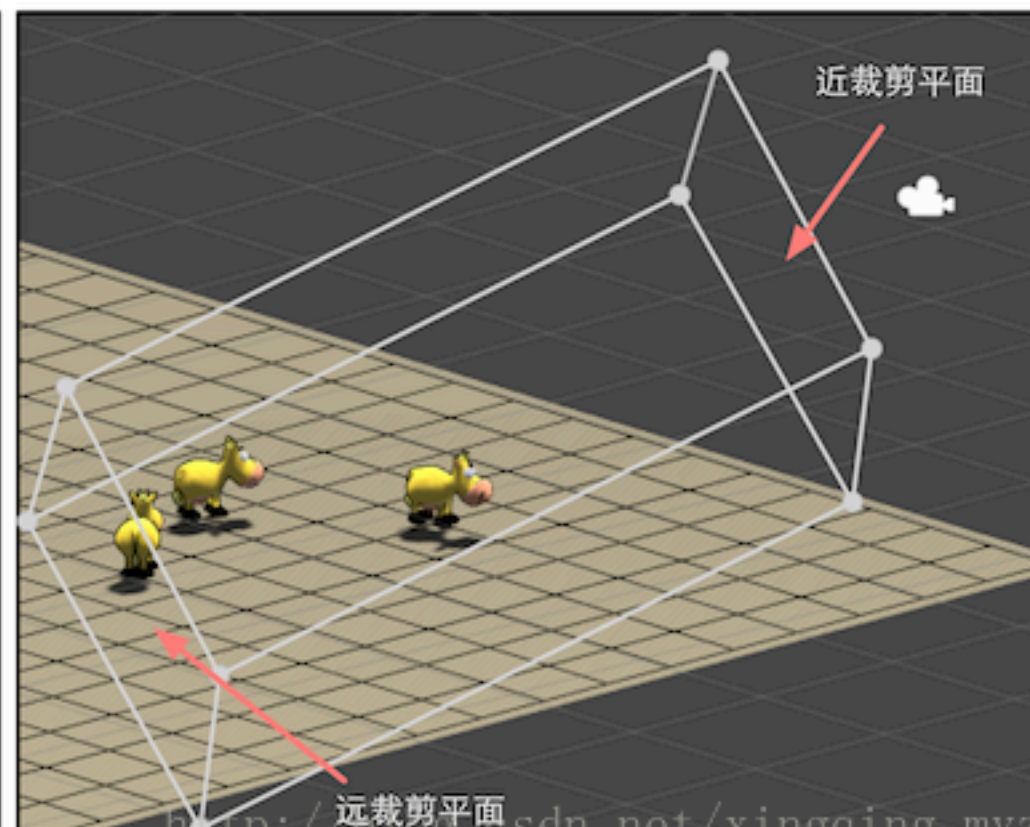


逻辑教育  
Logic education

### 3.投影



透视投影



平行投影

课程研发:CC老师  
课程授课:CC老师

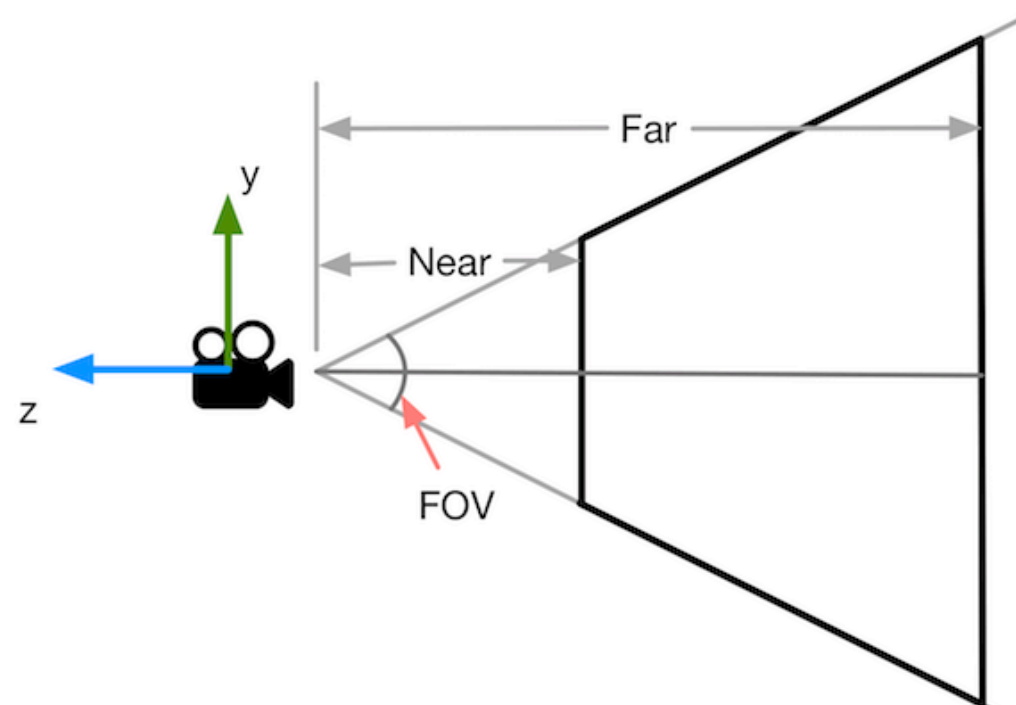
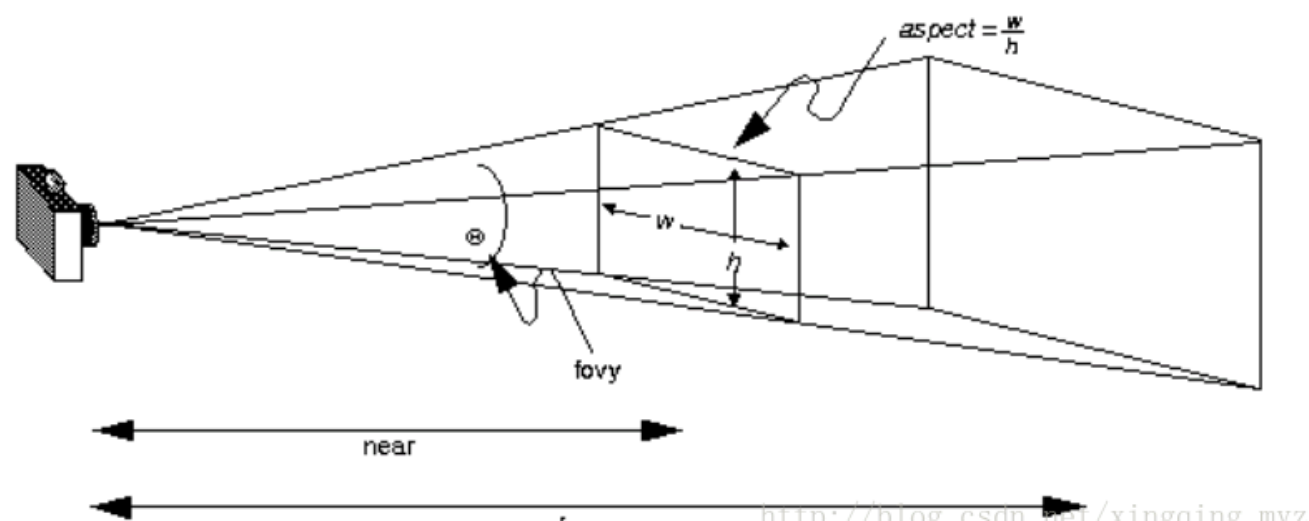
转载需注明出处,不得用于商业用途.已申请版权保护



逻辑教育  
Logic education

## 透视投影的视锥体

又称为，视景体、平截头体



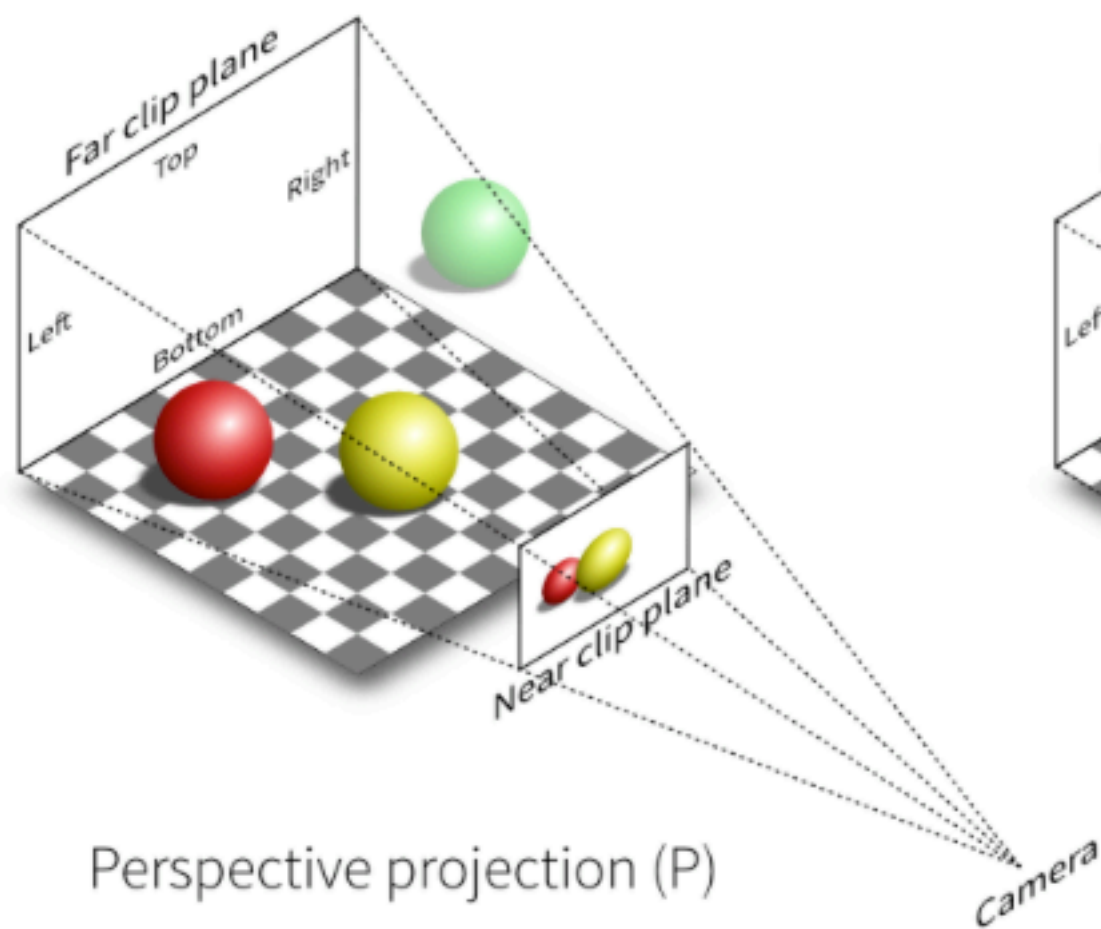
课程研发:CC老师  
课程授课:CC老师





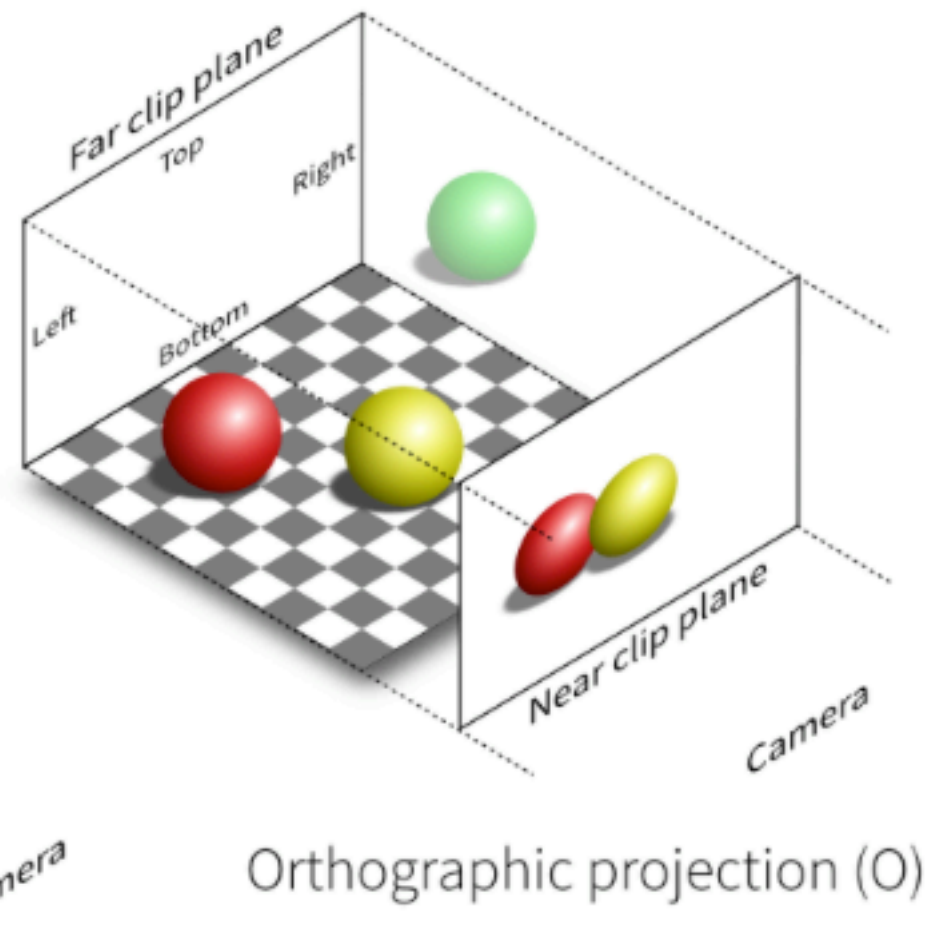
逻辑教育  
Logic education

## 平行投影



Perspective projection (P)

透视投影



Orthographic projection (O)

正投影

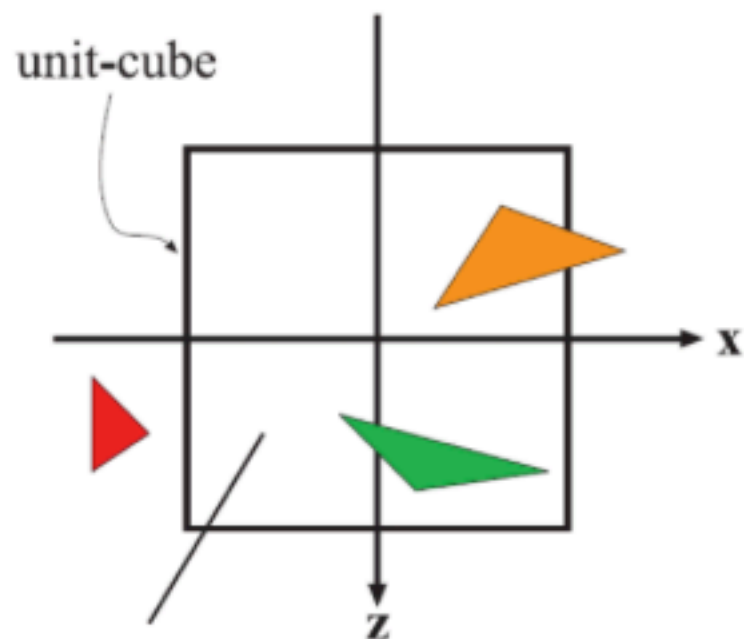
课程研发:CC老师  
课程授课:CC老师



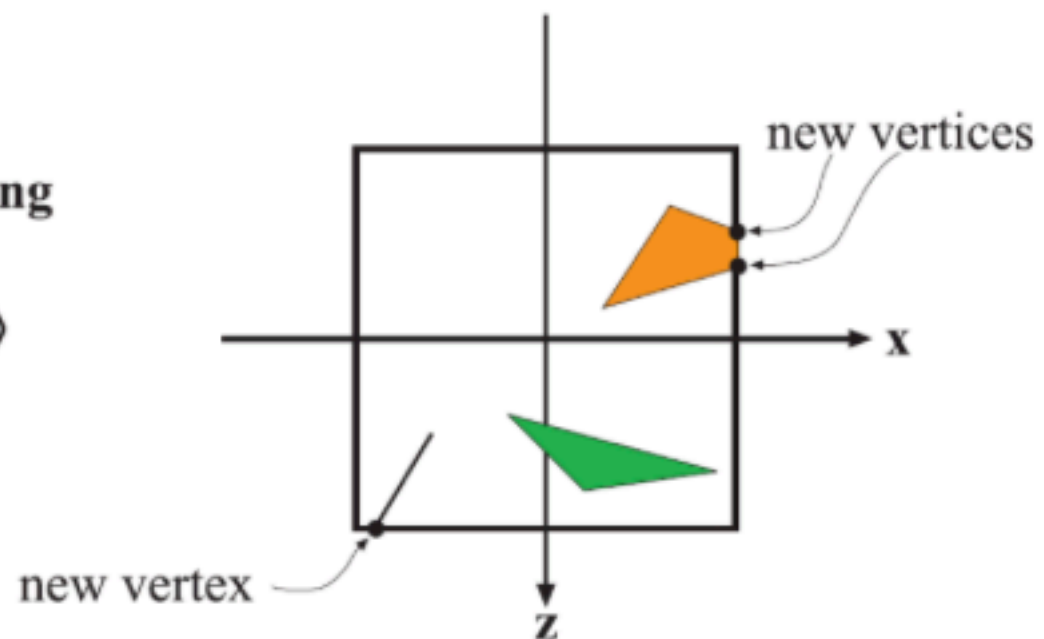
逻辑教育  
Logic education

## 裁剪

仅保留视景体内部的图元进行渲染



Clipping



课程研发:CC老师  
课程授课:CC老师



逻辑教育  
Logic education

## 屏幕映射

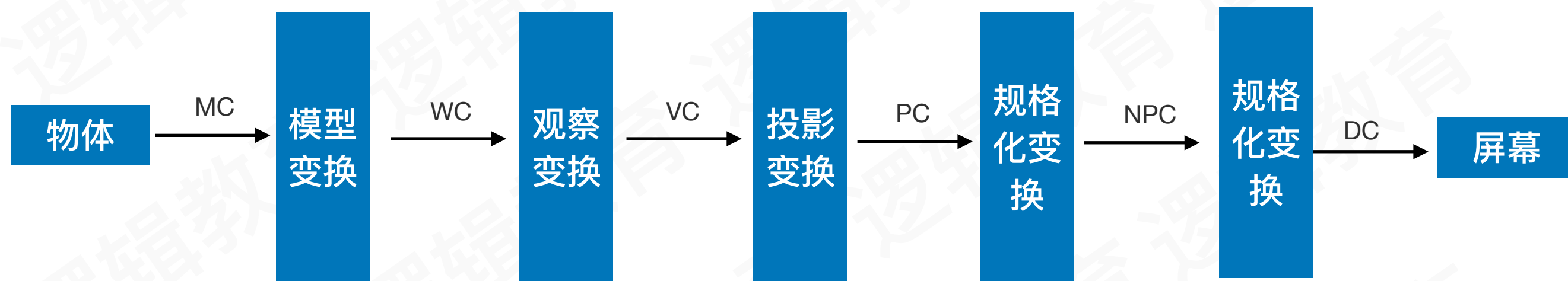
这个阶段是不可配置也不可编程的。主要用来实现图元的坐标转换到屏幕坐标

课程研发:CC老师  
课程授课:CC老师

转载需注明出处,不得用于商业用途.已申请版权保护



## 三维坐标转变为二维屏幕坐标的过程



三维观察流程

MC是建模坐标系，WC是世界坐标系，VC是观察坐标系，PC是投影坐标系，NPC是规格化投影坐标系，DC是设备坐标系

课程研发:CC老师  
课程授课:CC老师



## 坐标系详细概念

**世界坐标系** 坐标系统主要用于计算机图形场景中的所有图形对象的空间定位和定义

**局部坐标系** 独立于世界坐标系来定义物体几何特性

**观察坐标系** 观察坐标系通常是以视点的位置为原点，通过用户指定的一个向上的观察向量来定义整个坐标系统，观察坐标系主要用于从观察者的角度对整个世界坐标系内的对象进行重新定位和描述，从而简化几何物体在投影面的成像的数学推导和计算

**投影坐标系** 物体从世界坐标描述转换到观察坐标后，可将三维物体投影到二维表面上，即投影到虚拟摄像机的胶片上，这个过程就是投影变换。以胶片中心为参考原点的空间坐标系称为投影坐标系，物体在投影坐标系中的坐标称为投影坐标。

**设备坐标系** 是图形设备上采用的与具体设备相关的坐标系。设备坐标系一般采用整数坐标，其坐标范围由具体设备的分辨率决定。设备坐标系上的一个点一般对应图形设备上的一个像素。由于具体设备的限制，设备坐标系的坐标范围一般是有限的。

**规格化设备坐标系** 是为了避免设备相关性而定义的一种虚拟的设备坐标系。规格化坐标系的坐标范围一般从0到1，也有的是从-1到+1。采用规格化设备坐标系的好处是屏蔽了具体设备的分辨率，使得图形处理能够尽量避开对具体设备坐标的考虑。实际图形处理时，先将世界坐标转换成对应的规格化设备坐标，然后再将规格化设备坐标映射到具体的设备坐标上去。

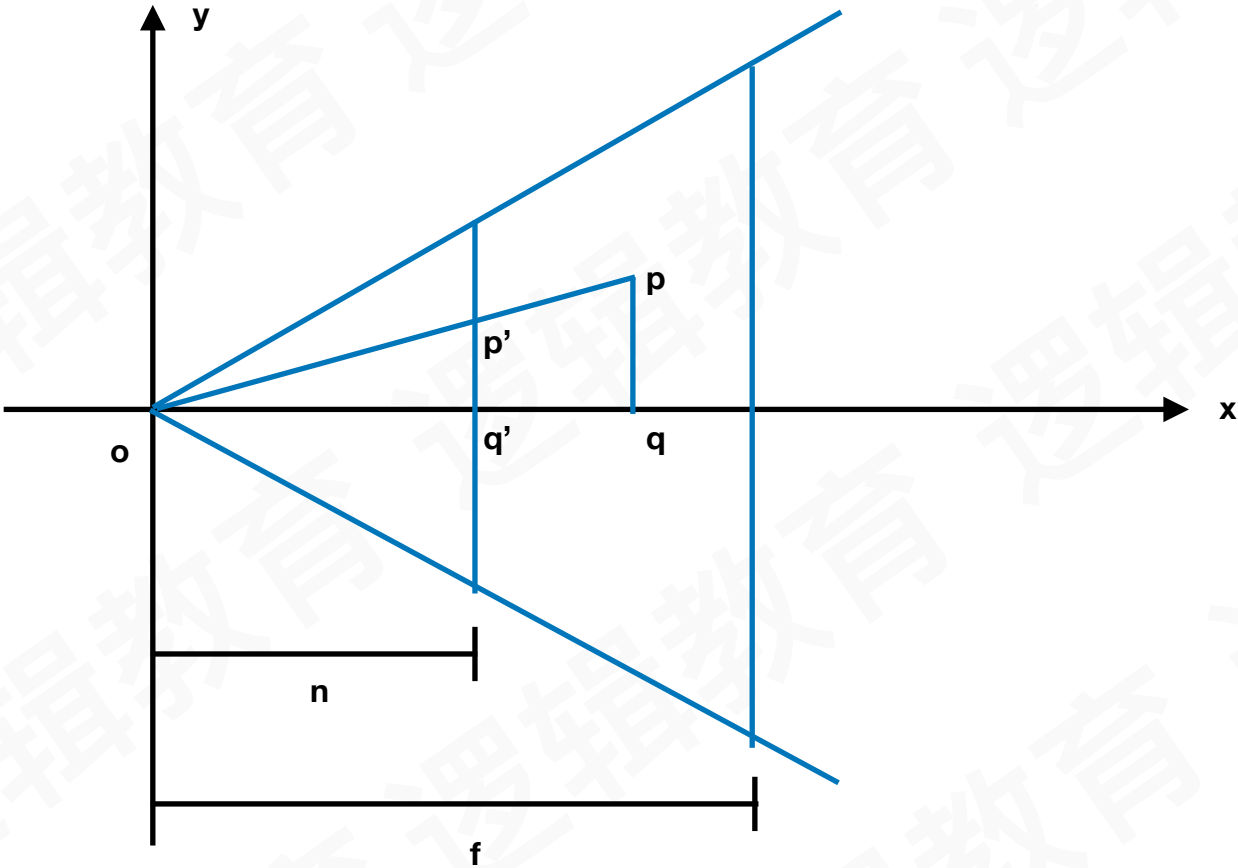
**屏幕坐标系统** 也称设备坐标系统，它主要用于某一特殊的计算机图形显示设备(如光栅显示器)的表面的点的定义，在多数情况下，对于每一个具体的显示设备，都有一个单独的坐标系统，在定义了成像窗口的情况下，可进一步在屏幕坐标系统中定义称为视图区(view port)的有界区域，视图区中的成像即为实际所观察到的。

课程研发:CC老师

课程授课:CC老师



逻辑教育  
Logic education



$n$ :近平面距离

$f$ :远平面距离

课程研发:CC老师  
课程授课:CC老师



逻辑教育  
Logic education

## GLSL 语法

- 变量和变量类型
- 向量和矩阵的构造以及选择
- 常量
- 结构和数组
- 运算符\控制流\函数
- 输入/输出变量,统一变量,统一变量块和布局限定符
- 预处理器和指令
- 精度修饰符与不变性

课程研发:CC老师  
课程授课:CC老师



## GLSL 变量与变量类型

### 向量数据类型

类型	描述
vec2,vec3,vec4	2分量、3分量、4分量浮点向量
ivec2,ivec3,ivec4	2分量、3分量、4分量整型向量
uvec2,uvec3,uvec4	2分量、3分量、4分量无符号整型向量
bvec2,bvec3,bvec4	2分量、3分量、4分量bool型向量



# GLSL 变量与变量类型

## 矩阵数据类型

类型	描述
mat2,mat2x2	两行两列
mat3,mat3x3	三行三列
mat4,mat4x4	四行四列
mat2x3	三行两列
mat2x4	四行两列
mat3x2	两行三列
mat3x4	四行三列
mat4x2	两行四列
mat4x3	三行四列

## mat列x行

课程研发:CC老师  
课程授课:CC老师



## GLSL 变量与变量类型

### 变量存储限定符

限定符	描述
<none>	只是普通的本地变量，外部不见，外部不可访问
const	一个编译常量，或者说是一个对函数来说为只读的参数
in/varying	从以前阶段传递过来的变量
in/varying centroid	一个从以前的阶段传递过来的变量，使用质心插值
out/attribute	传递到下一个处理阶段或者在一个函数中指定一个返回值
out/attribute centroid	传递到下一个处理阶段，质心插值
uniform	一个从客户端代码传递过来的变量，在顶点之间不做改变

质心插值学术文献: <http://www.ixueshu.com/document/f2f5be57efaaad68.html>

课程研发:CC老师  
课程授课:CC老师



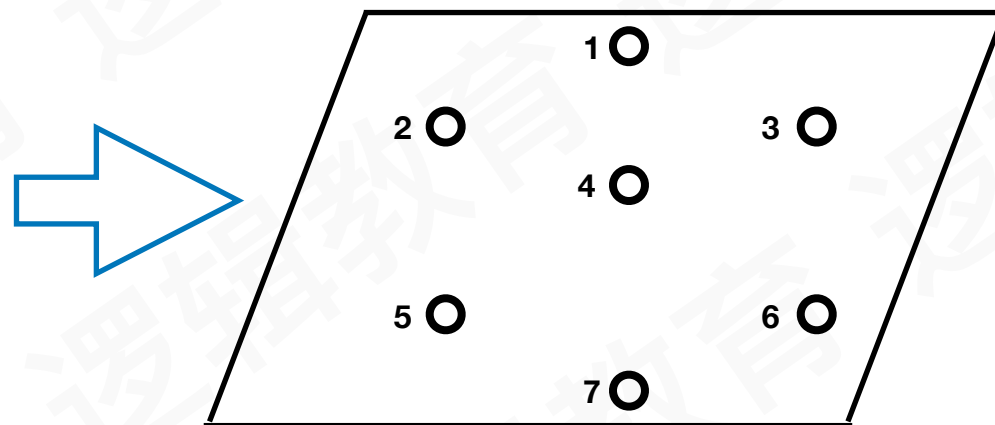
## 2.生成图元

顶点数组

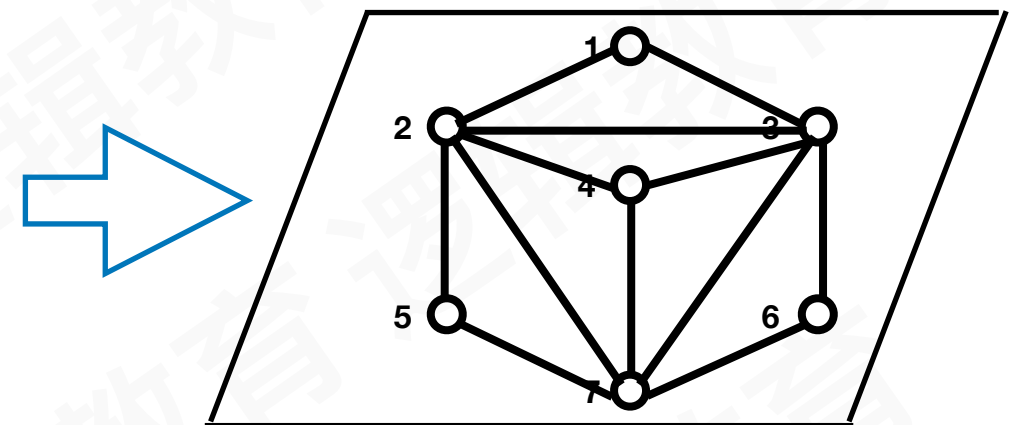


索引数组 {1,2,3}  
{3,2,4}  
{4,2,7}  
{7,2,5}

顶点着色程序



三角形组合

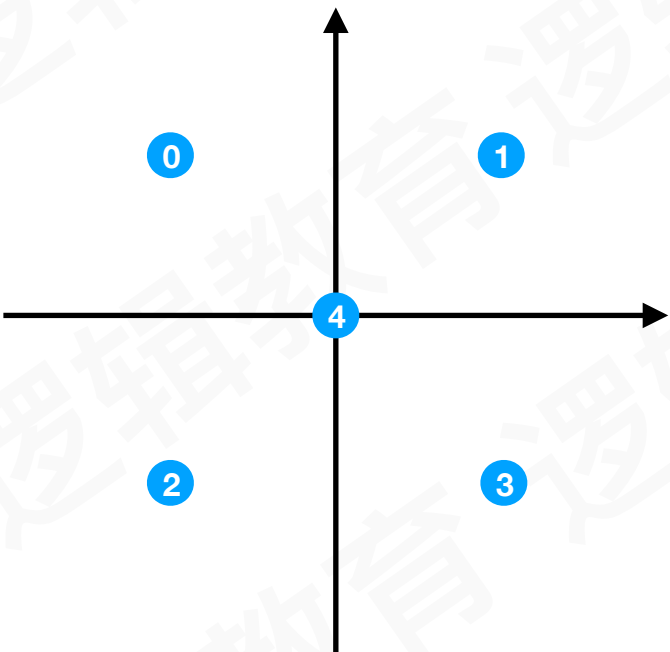


课程研发:CC老师  
课程授课:CC老师

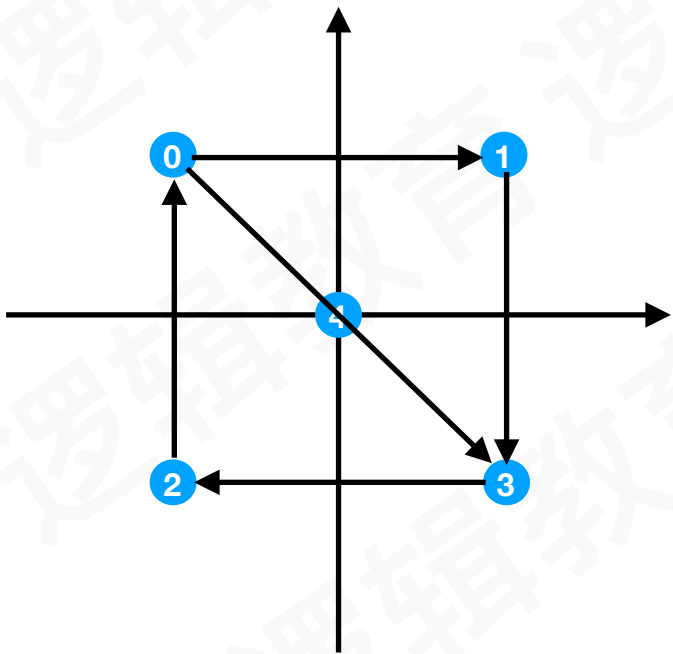




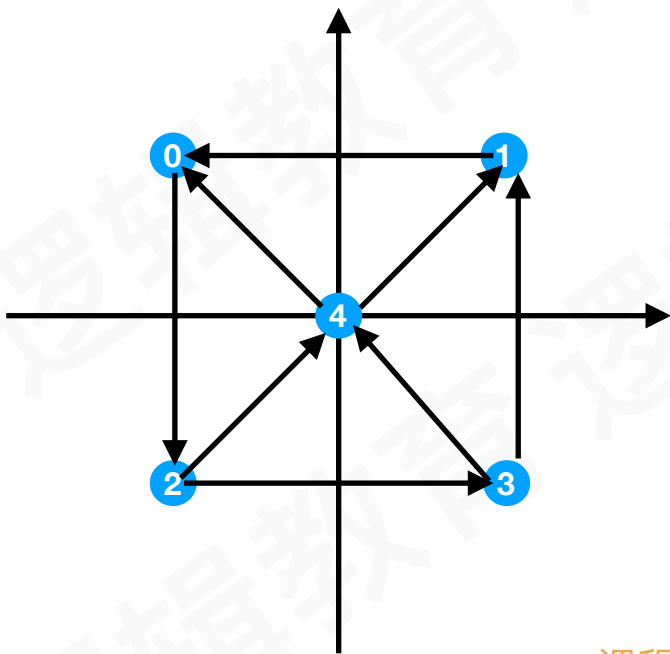
索引绘图



- 0  $-0.5f, 0.5f, 0.0f,$
- 1  $0.5f, 0.5f, 0.0f,$
- 2  $-0.5f, -0.5f, 0.0f,$
- 3  $0.5f, -0.5f, 0.0f,$
- 4  $0.0f, 0.0f, 1.0f,$



0 3 2  
0 1 3  
金字塔底部



0, 2, 4,  
0, 4, 1,  
2, 3, 4,  
1, 4, 3,  
金字塔四侧

课程研发:CC老师  
课程授课:CC老师

