

经济危机
创业

1万元如何创业



win7旗舰版

学习op
eng

opengl投



小本创业

创业新点
子

冷门创业好项目



方海龙的图书馆 图书馆



53 馆藏 571

OpenGL学习脚印：投影矩阵的推导

2014-10-28 方海龙的... 摘自 csdn博客 阅 1863 转 17

分享： 微信 转藏到我的图书馆

OpenGL学习脚印: 投影矩阵的推导

写在前面

本节内容翻译和整理自<http://www.songho.ca> songho的博客《[OpenGL Projection Matrix](#)》内容,以供自己和初学者熟悉投影矩阵推导过程。

通过本节，你可以了解到：

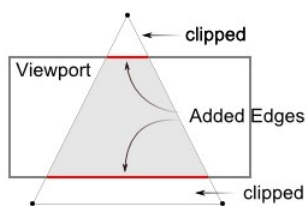
投影矩阵计算的阶段

透视投影和正交投影的矩阵推导

本节的要点就在于：阅读时，自己拿笔推导一遍。

1.概览

计算机屏幕是2维的，OpenGL渲染的3D场景必须以2D形式的图像投影到屏幕上。GL_PROJECTION 矩阵就是用来设置投影变换的。首先，它将所有顶点从眼坐标(照相机坐标)转换到裁剪坐标系下。然后，这些裁剪坐标通过透视除法，即除以裁剪坐标中w分量，转换到归一化设备坐标系(NDC)。



一个由视锥裁剪的三角形

因此，我们要记住，裁剪(视锥剔除frustum culling)和NDC转换都集成到了GL_PROJECTION 矩阵。接下来的部分描述了怎么样通过left, right, bottom, top, near and far 这6个界限参数来构造投影矩阵。

注意：

视锥剔除是在裁剪坐标系中进行的，并且恰好在透视除法之前进行。裁剪坐标xc, yc 和 zc 通过与wc比较来进行测试。如果某个坐标值比wc小或者比wc大，那么这个顶点将被丢弃。然后，OpenGL会重新在裁剪进行的地方构造多边形的边缘。

补充内容：

实际上，眼坐标系下坐标在乘以投影矩阵后，裁剪测试和透视除法都是由GPU来执行的。而后面这两个过程处理的裁剪坐标系数据都是由投影矩阵变换的。

1. 裁剪测试也即视锥剔除

$-W_c < X_c, Y_c, Z_c < W_c$

2. NDC透视除法

$X_n = X_c / W_c \quad Y_n = Y_c / W_c \quad Z_n = Z_c / W_c$

这里需要注意的是，我们在构造16个参数的投影矩阵的同时，不仅要考虑到裁剪，还要考虑到透视除法的过程。这样，最终的NDC坐标才会满足：

$-1 < X_n, Y_n, Z_n < 1$

TA的最新馆藏

写出高效清晰Layout布局文件的一...

专题：Android 各类优化方案大全

C++中map、hash_map、unorde..

4. 建造者模式

HTTP Keep

深入理解HTTP协议

推荐阅读

OpenGL学习脚印: 静止的物体

OpenGL学习脚印: 变换法向量 Tra...

OpenGL学习脚印: 理解坐标系及坐...

OpenGL学习脚印: OpenGL 坐标...

OpenGL学习脚印: 理解坐标系及坐...

OpenGL学习脚印: 什么是OpenGL?

OpenGL学习错误列表Error List

OpenGL学习脚印: 模型变换(mode...

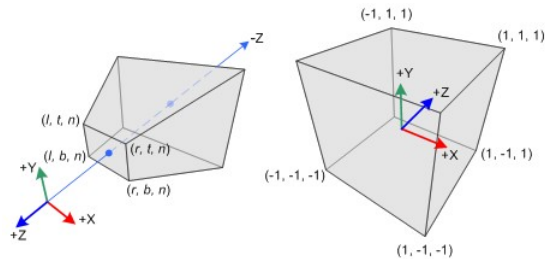
OpenGL学习脚印: 顶点数据传送和...

1 投影仪维修电话	7 地面互动投影
2 互动投影	8 投影仪家用
3 互动投影系统	9 地面互动
4 视频矩阵品牌	10 高清矩阵
5 家用投影机	11 投影仪
6 学习英语单词	12 高清混合矩阵

在透视投影中，在眼坐标下截头锥体(a truncated pyramid frustum)内的3D点被映射到NDC下一个立方体中；x坐标从 $[l, r]$ 映射到 $[-1, 1]$ ，y坐标从 $[b, t]$ 映射到 $[-1, 1]$ ，z坐标从 $[n, f]$ 映射到 $[-1, 1]$ 。

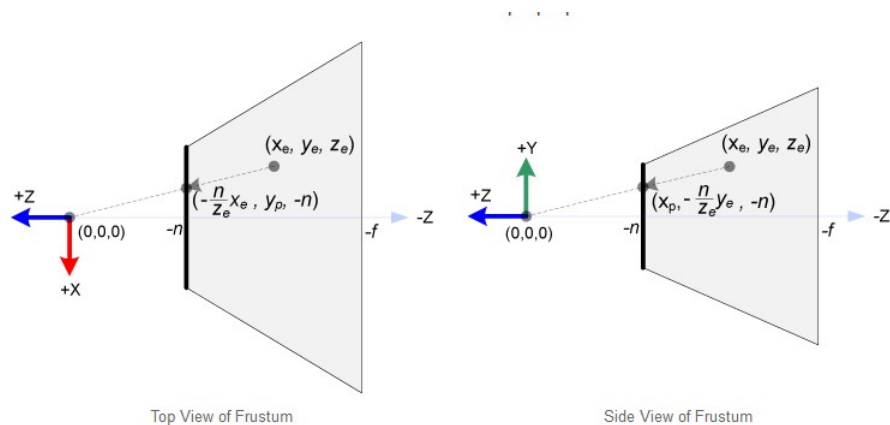
注意:

眼坐标系使用右手坐标系，而NDC使用左手坐标系。这就是说，眼坐标系下，在 origin 处的照相机朝着-Z轴看去，但是在NDC中它朝着+Z轴看去。因为`glFrustum()` 仅接受正的near和far距离，我们在构造`GL_PROJECTION` 矩阵时，需要取其相反数。眼坐标系和NDC坐标系如下图所示:



Perspective Frustum and Normalized Device Coordinates (NDC)

在OpenGL中,眼坐标下3D点被投影到近裁剪面(即投影平面)。下图展示了眼坐标系下点 (x_e, y_e, z_e) 如何投影到近裁剪面上的 (x_p, y_p, z_p) 的。(左侧是视锥的俯视图，右侧是视锥的侧视图,拿出右手构成右手坐标系，然后比划比划就出来了)



根据三角形的相似性，由俯视图可得出:

$$\begin{aligned} \frac{x_p}{x_e} &= \frac{-n}{z_e} \\ x_p &= \frac{-n \cdot x_e}{z_e} = \frac{n \cdot x_e}{-z_e} \end{aligned}$$

由侧视图可以得出:

$$\begin{aligned} \frac{y_p}{y_e} &= \frac{-n}{z_e} \\ y_p &= \frac{-n \cdot y_e}{z_e} = \frac{n \cdot y_e}{-z_e} \end{aligned}$$

补充: x_p 和 y_p 其实是一个中间值，我们要找的是 (x_c, y_c, z_c) 和 (x_n, y_n, z_n) 之间的关系，但是可以利用:

$$\begin{aligned} x_p &= \frac{n x_e}{-z_e} \\ y_p &= \frac{n y_e}{-z_e} \\ z_p &= -n \end{aligned}$$

这一关系做过渡，后面利用 x_p 和 y_p ，映射到NDC中 x_n 和 y_n 的线性关系就利用到了 x_p 和 y_p 。这一点很重要。



小本创业开店



众筹创业

经济危机 创业

学习opengl

电脑重装系统

opengl投影

经济危机 商机

冷门创业好项目

关闭

于 z_e ，他们与 $-z_e$ 成反比。换句话说，他们都被 $-z_e$ 相除。这个事构造GL_PROJECTION 矩阵时取值的线索。在眼坐标通过乘以GL_PROJECTION 未转换时，裁剪坐标系仍然是一个齐次坐标系。通过对裁剪坐标进行透视除法得到最终的NDC坐标。下图解释了这个过程：

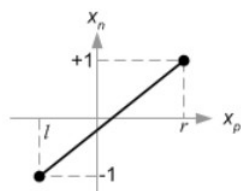
$$\begin{pmatrix} x_{clip} \\ y_{clip} \\ z_{clip} \\ w_{clip} \end{pmatrix} = M_{projection} \cdot \begin{pmatrix} x_{eye} \\ y_{eye} \\ z_{eye} \\ w_{eye} \end{pmatrix}, \quad \begin{pmatrix} x_{ndc} \\ y_{ndc} \\ z_{ndc} \end{pmatrix} = \begin{pmatrix} x_{clip}/w_{clip} \\ y_{clip}/w_{clip} \\ z_{clip}/w_{clip} \end{pmatrix}$$

因此我们可以把裁剪坐标系下的w分量设为 $-z_e$ ，那么GL_PROJECTION矩阵第4行变为(0, 0, -1, 0), 如下图(求出了投影矩阵第4行):

$$\begin{pmatrix} x_c \\ y_c \\ z_c \\ w_c \end{pmatrix} = \begin{pmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} x_e \\ y_e \\ z_e \\ w_e \end{pmatrix}, \quad \therefore w_c = -z_e$$

现在我们把 x_p 和 y_p ，映射到NDC中 x_n 和 y_n ，他们之间是线性关系: [l, r] ? [-1, 1]和[b, t] ? [-1, 1].

线性关系如下图所示:



Mapping from x_p to x_n

则可以推导出:

$$x_n = \frac{1-(-1)}{r-l} \cdot x_p + \beta$$

$$1 = \frac{2r}{r-l} + \beta \quad // \text{ substitute } (x_p, x_n) \text{ with } (r, 1)$$

$$\beta = 1 - \frac{2r}{r-l} = \frac{r-l}{r-l} - \frac{2r}{r-l} = \frac{r-l-2r}{r-l} = \frac{-r-l}{r-l}$$

$$= -\frac{r+l}{r-l}$$

$$\therefore x_n = \frac{2x_p}{r-l} - \frac{r+l}{r-l}$$

细节部分有删节,这个推导过程使用的就是简单的 $y=kx+b$ 线性关系推导，同理利用[b, t] ? [-1, 1]可推得:

$$y_n = \frac{2y_p}{t-b} - \frac{t+b}{t-b}$$

将上面的 x_p 和 y_p 带入求得:

$$x_n = \frac{2x_p}{r-l} - \frac{r+l}{r-l} \quad \left(x_p = \frac{nx_e}{-z_e} \right)$$

$$= \frac{2 \cdot \frac{nx_e}{-z_e}}{r-l} - \frac{r+l}{r-l}$$

$$= \frac{2n \cdot x_e}{(r-l)(-z_e)} - \frac{r+l}{r-l}$$

$$= \frac{2n}{r-l} \cdot \frac{x_e}{-z_e} - \frac{r+l}{r-l}$$

$$= \frac{2n}{r-l} \cdot \frac{x_e}{-z_e} + \frac{r+l}{r-l} \cdot \frac{z_e}{-z_e}$$

$$= \left(\frac{2n}{r-l} \cdot x_e + \frac{r+l}{r-l} \cdot z_e \right) / -z_e$$

$$y_n = \frac{2y_p}{t-b} - \frac{t+b}{t-b} \quad \left(y_p = \frac{ny_e}{-z_e} \right)$$

$$= \frac{2 \cdot \frac{ny_e}{-z_e}}{t-b} - \frac{t+b}{t-b}$$

$$= \frac{2n \cdot y_e}{(t-b)(-z_e)} - \frac{t+b}{t-b}$$

$$= \frac{2n}{t-b} \cdot \frac{y_e}{-z_e} - \frac{t+b}{t-b}$$

$$= \frac{2n}{t-b} \cdot \frac{y_e}{-z_e} + \frac{t+b}{t-b} \cdot \frac{z_e}{-z_e}$$

$$= \left(\frac{2n}{t-b} \cdot y_e + \frac{t+b}{t-b} \cdot z_e \right) / -z_e$$

注意这里 x_n 和 y_n 已经是NDC中的坐标了，通过这两个坐标可以求出GL_PROJECTION 的前两行来，书写如下(求出了投影矩阵第1,2,4行):

$$\begin{pmatrix} y_c \\ z_c \\ w_c \end{pmatrix} = \begin{pmatrix} 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ . & . & . & . \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} x_e \\ y_e \\ z_e \\ w_e \end{pmatrix}$$

现在怎么求出第3行呢？

找出 z_n 与找出 x_n 和 y_n 不同，因为 z_e 总是被投影近裁剪面-n上。但是我们需要唯一的Z值进行裁剪和深度测试。另外，我们还能够unproject即反向变换(inverse transform)。因为Z值不依赖于x或者y，因此我们借用w分量来找出 z_n 和 z_e 之间的关系。

因此我们可以这样指定第3行：

$$\begin{pmatrix} x_c \\ y_c \\ z_c \\ w_c \end{pmatrix} = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & A & B \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} x_e \\ y_e \\ z_e \\ w_e \end{pmatrix}, \quad z_n = \frac{z_c}{w_c} = \frac{Az_e + Bw_e}{-z_e}$$

在眼坐标下 w_e 等于1，因此上式变为：

$$z_n = \frac{Az_e + B}{-z_e}$$

我们使用 (z_e, z_n) 的关系 $(-n, -1)$ 和 $(-f, 1)$ 来求解出系数A,B；

$$\begin{cases} \frac{-An+B}{-n} = -1 \\ \frac{-Af+B}{-f} = 1 \end{cases} \rightarrow \begin{cases} -An + B = -n & (1) \\ -Af + B = f & (2) \end{cases}$$

细节部分有删节，使用消元法即可求出：

$$A = -\frac{f+n}{f-n} \quad B = -\frac{2fn}{f-n}$$

我们求出了A和B，那么 z_e 和 z_n 关系如下式：

$$z_n = \frac{-\frac{f+n}{f-n}z_e - \frac{2fn}{f-n}}{-z_e} \quad (3)$$

最终的投影矩阵如下式：

$$\begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

OpenGL Perspective Projection Matrix

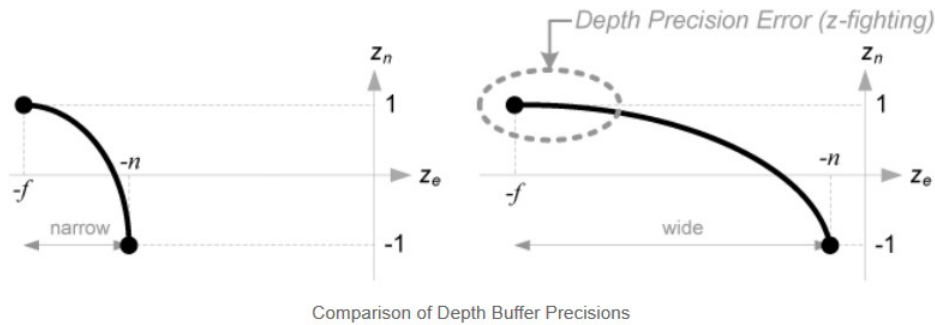
这个公式对应的是一般的视锥，如果视锥是对称的，即 $r = -l, t = -b$ ，那么有：

$$\begin{cases} r+l=0 \\ r-l=2r \text{ (width)} \end{cases}, \quad \begin{cases} t+b=0 \\ t-b=2t \text{ (height)} \end{cases}$$

$$\begin{pmatrix} \frac{n}{r} & 0 & 0 & 0 \\ 0 & \frac{n}{t} & 0 & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

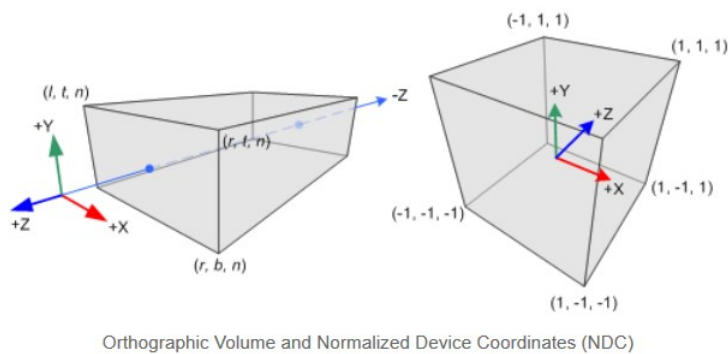
z-fighting

表示 z_e 和 z_n 关系的式2,这是一个有理函数,并且 x_e 和 z_e 之间不是线性关系。这意味着,在近裁剪面的精度很高,而在远裁剪面则很小。如果 $[-1, 1]$ 范围变得太大,就会引起深度精度问题,即z-fighting。在远裁剪面附近, z_e 的小变化根本不影响 z_n 值。近裁剪面和远裁剪面之间的 n 和 f 举例,应该尽可能小,来减少深度缓存的精度问题。可参考下图来帮助理解。



3.正交投影

构造正交投影的矩阵简单很多。所有的是眼坐标 x_e, y_e 和 z_e ,都被线性的映射到NDC中。我们需要做的就是讲长方体视锥体缩放为规范视见体,然后移动到原点。如下图所示:



以 x_e 和 x_n 之间映射关系为例, $[l, r] = [-1, 1]$,则可以推导如下:

$$x_n = \frac{1 - (-1)}{r - l} \cdot x_e + \beta$$

$$1 = \frac{2r}{r - l} + \beta \quad // \text{ substitute } (x_e, x_n) \text{ with } (r, 1)$$

$$\beta = 1 - \frac{2r}{r - l} = -\frac{r + l}{r - l}$$

$$\therefore x_n = \frac{2}{r - l} \cdot x_e - \frac{r + l}{r - l}$$

Mapping from x_e to x_n

y, z 也有类似推导, 这里省略, 最后得出投影矩阵为:

$$\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & -\frac{r+l}{r-l} \\ 0 & \frac{2}{t-b} & 0 & -\frac{t+b}{t-b} \\ 0 & 0 & \frac{-2}{f-n} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

OpenGL Orthographic Projection Matrix

如果视锥是对称的话, 即 $r = -l, t = -b$ 的话, 则可以简化为:

$$\begin{cases} r-l=2r & (\text{width}) \\ t-b=2t & (\text{height}) \end{cases}$$

$$\begin{cases} t+b=0 \\ t-b=2t \end{cases}$$

$$\begin{pmatrix} \frac{1}{r} & 0 & 0 & 0 \\ 0 & \frac{1}{t} & 0 & 0 \\ 0 & 0 & \frac{-2}{f-n} & -\frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

到这里透视投影和正交投影矩阵推导完毕。

转藏到我的图书馆

献花 (0)

分享：

微信

来自：方海龙的图书馆 > 《图形学》

以文找文 | 举报

上一篇：OpenGL学习脚印: OpenGL 坐标变换

下一篇：OpenGL学习脚印: 绘制移动三角形

猜你喜欢

1万元如何创业 opengl投影 经济危机 创业 年轻人如何创业 学习opengl
经济危机 房贷 创业 创业好点子 创业新点子 众筹创业 办厂创业好项目
opengl入门学习 创业好项目 冷门创业好项目 opengl怎么样 电脑重装系统
投影矩阵推导 opengl投影矩阵 投影矩阵 学习矩阵 opengl透视投影 如何创业
opengl学习笔记 opengl投影方式 透视投影矩阵 如何学习opengl 正交投影
矩阵算法 小本创业开店 互联网创业 农村创业好项目 音视频矩阵
1万元如何创业 opengl投影 经济危机 创业 年轻人如何创业 学习opengl

类似文章

更多

精选文章

图形中投影矩阵的推导

3D数学知识简介

特征值与特征向量：信号处理中的应用

LDA?线性判别分析

摄像机成像中的若干重要空间关系

空间直角坐标转换之仿射变换 | IT爱好者...

摄像机标定

理解计算机3D图形学中的坐标系变换

中国俗语、谚语大全和解释

十大高智商电影

他改变了世界——戈尔巴乔夫：我失去权力，...

小学数学常用公式一览表

八一八海军三大舰队的六大驱支

炒肉时如何把肉炒得更软

80首古典音乐

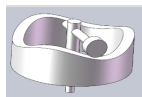
周易的卦与爻



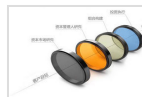
经济危机 创业



创业好项目



solidworks放大镜



怎么学习投资?



PPT课件制作9:平



大资金秘诀 波浪



【两学一做】学习



最全的厨房管理..



矩阵lu分解求逆详



做一个会学习,会

- 1 惊!让记忆力提高100万倍
- 2 财富向你招手 百万富翁不是..
- 3 财富向你招手 百万富翁不是..

- 1 投影仪维修电话
- 2 互动投影
- 3 互动投影系统
- 4 投影仪家用
- 5 高清混合矩阵
- 6 学习英语单词

发表评论：

请 [登录](#) 或者 [注册](#) 后再进行评论

社交帐号登录：