



计算机图形学小白入门

——从0开始实现OpenGL

摄像机系统设计

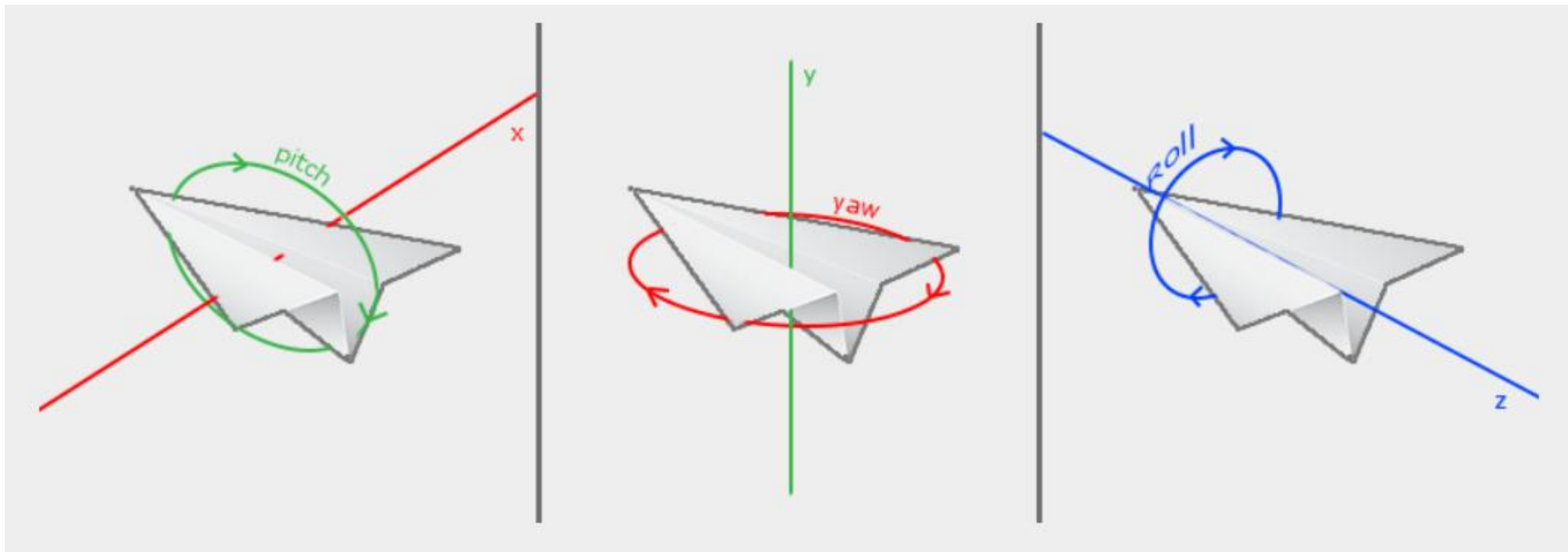


授课：赵新政
资深三维工程师

专注3D图形学技术
教育品牌

摄像机使用

- WASD：代表**前平移/左平移/后平移/右平移**
- 鼠标右键：点住左右移动——左右观望；点住上下平移——上下点头

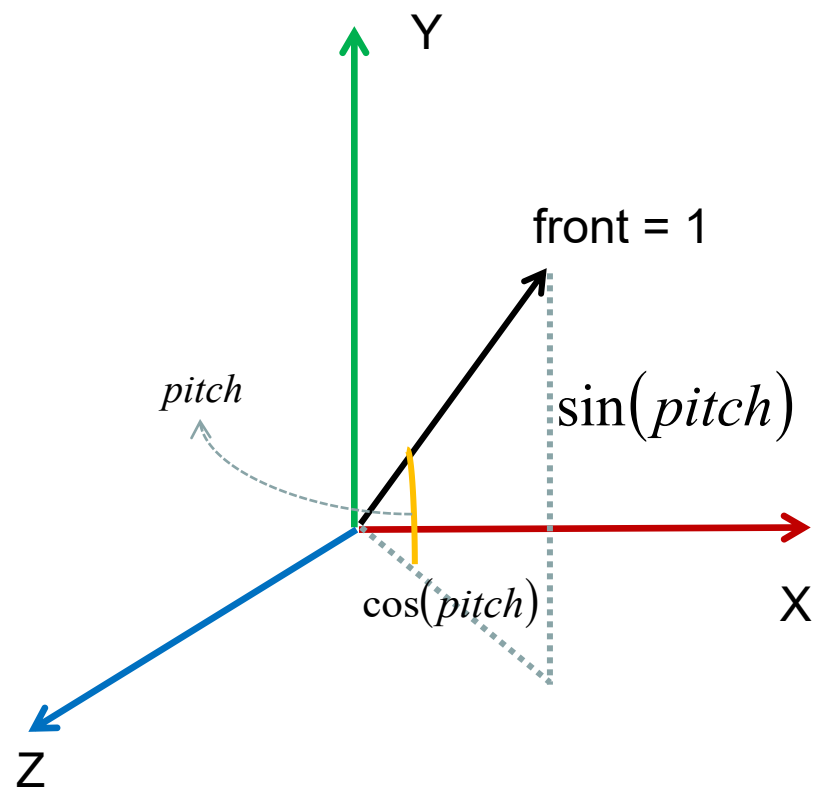


摄像机设计

Camera

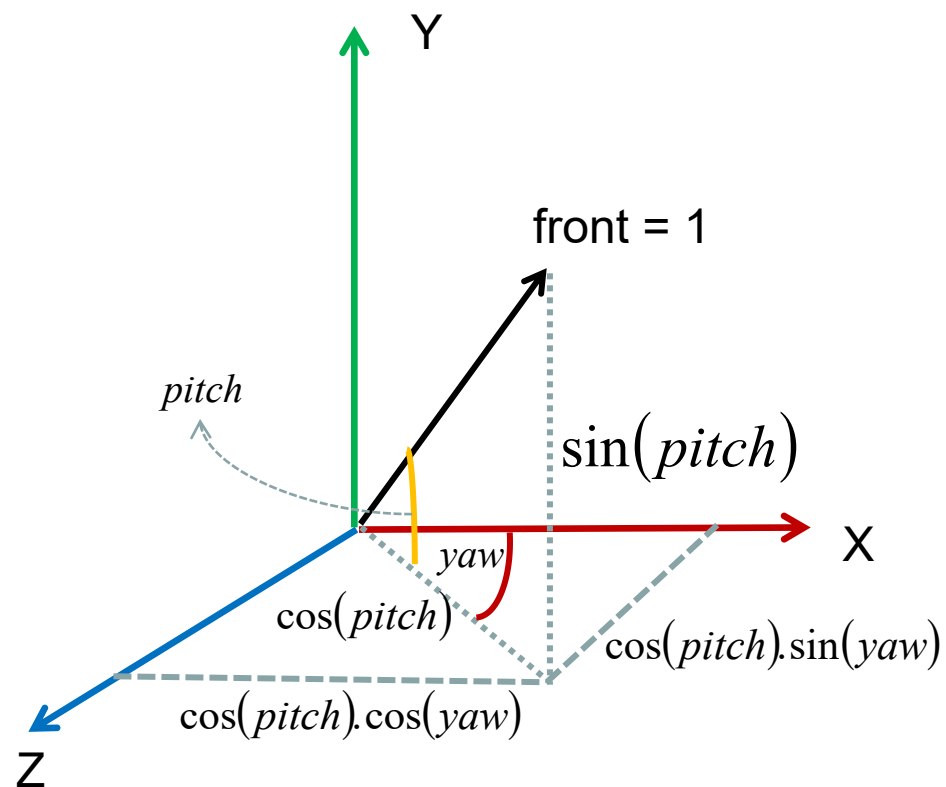
math::vec3f	mPosition{ 0.0f,0.0f,0.0f };	//当前位置
math::vec3f	mFront{ 0.0f,0.0f,-1.0f };	//摄像机空间方向
math::vec3f	mTop{ 0.0f,1.0f,0.0f };	//摄像机穹顶向量
float	mSpeed = { 0.01f };	//摄像机移动速度
float	mPitch{ 0.0f };	//摄像机当前仰角
float	mYaw{ -90.0f };	//摄像机绕y轴旋转角度
float	mSensitivity{ 0.1f };	//鼠标滑过屏幕距离对旋转角影响敏感度

摄像机旋转—front的y值



$$front.y = \sin(pitch)$$

摄像机旋转—front的y值

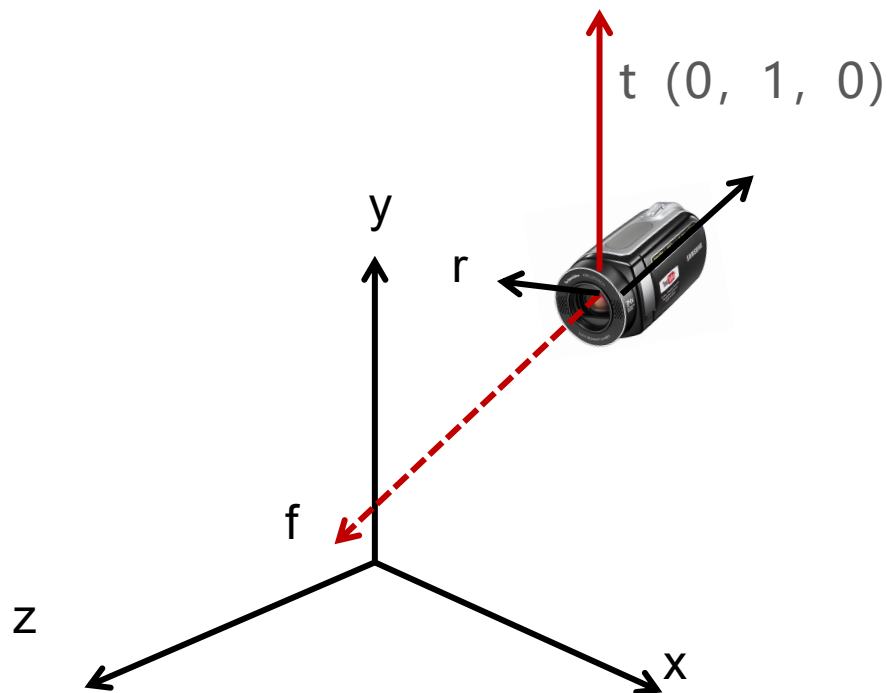


$$\begin{aligned} \text{front}.y &= \sin(\text{pitch}) \\ \text{front}.z &= \cos(\text{pitch}) \cdot \sin(\text{yaw}) \\ \text{front}.x &= \cos(\text{pitch}) \cdot \cos(\text{yaw}) \end{aligned}$$

构建摄像机矩阵（回顾）

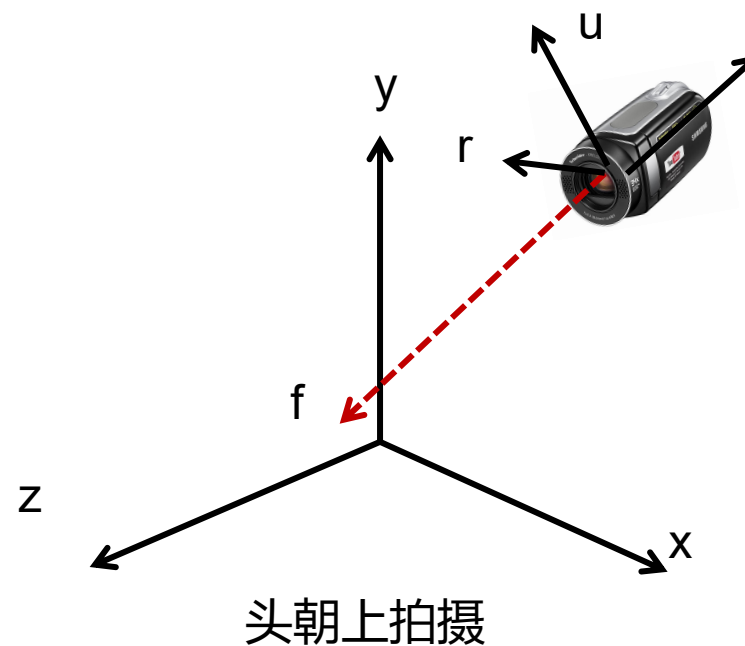
- 顶构建右侧向量 r 与头顶向量 u
- 已知方向向量 f 以及顶部向量 t , 可以知道:

$$\vec{r} = \text{normalize}(\vec{f} \times \vec{t})$$



- 已知方向向量 f 以及右部向量 r , 可以知道:

$$\vec{u} = \vec{r} \times \vec{f}$$



LookAt函数

- 又知ViewMatrix的构成如下

$$viewMatrix = \begin{pmatrix} r_x & r_y & r_z & -\vec{r} \cdot \vec{p} \\ u_x & u_y & u_z & -\vec{u} \cdot \vec{p} \\ -f_x & -f_y & -f_z & \vec{f} \cdot \vec{p} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- 在MathFunction中加入如下函数:

```
mat4f lookAt(vec3 eye, vec3 center, vec3 top)
```

eye: 当前摄像机位置

center: 摄像机看向哪个空间中的点

top: 摄像机穹顶向量