

开发笔记

Ivan Lin

2017 年 1 月 25 日

Visual Studio

Resharper插件

alt + o: .h和.cpp文件切换

alt + 鼠标: 框选模式

ctrl+k + ctrl+c: 注释代码

shift+alt+up/down: 框选模式上下

ctrl+alt+a: open Command Window

ReSharper.Suspend/ReSharper.Resume in Command Window: close/open

ReSharper

lib:静态库, 在编译时将库代码加入程序中; dll:动态库, 编译时生成一个lib和一个dll, lib用于存放dll中相应接口的索引

计算机图形学

坐标系模拟: 拇指x, 食指y, 中指z。左手系和右手系

标准化向量= 单位向量= 法线, $\mathbf{v}_{norm} = \frac{\mathbf{v}}{|\mathbf{v}|}$

$\mathbf{a} + \mathbf{b}$ 几何解释: \mathbf{a} 的头连接 \mathbf{b} 的尾, 然后从 \mathbf{a} 的尾向 \mathbf{b} 的头画一个向量

$\mathbf{a} - \mathbf{b}$ 几何解释: \mathbf{a} 的尾连接 \mathbf{b} 的尾, 然后从 \mathbf{b} 的头向 \mathbf{a} 的头画一个向量

向量点乘: $\mathbf{a} \cdot \mathbf{b} (\mathbf{ab}) = a_1b_1 + \dots + a_nb_n$, 几何解释: $\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}|\cos\theta$ (两向量夹角)

向量投影: \mathbf{v} 分解为平行和垂直于 \mathbf{n} 的两个分量。

$$\mathbf{v}_{\parallel} = \mathbf{n} \frac{\mathbf{v} \cdot \mathbf{n}}{|\mathbf{n}|^2} \quad \mathbf{v}_{\perp} = |\mathbf{v}| - \mathbf{v}_{\parallel}$$

向量叉乘：仅可用于3D向量， $\mathbf{a} \times \mathbf{b} = \begin{bmatrix} \mathbf{a}_y \mathbf{b}_z - \mathbf{a}_z \mathbf{b}_y \\ \mathbf{a}_z \mathbf{b}_x - \mathbf{a}_x \mathbf{b}_z \\ \mathbf{a}_x \mathbf{b}_y - \mathbf{a}_y \mathbf{b}_x \end{bmatrix}$ ，几何解释：结果向

量垂直于原来两个向量， $|\mathbf{a} \times \mathbf{b}| = |\mathbf{a}||\mathbf{b}|\sin\theta$

， $|\mathbf{a} \times \mathbf{b}| = 0$ 表示 \mathbf{a} 与 \mathbf{b} 平行或有一个为 $\mathbf{0}$

矩阵转置： M^T ，其列由 \mathbf{M} 的行组成， $M^T_{ji} = M_{ij}$

$(AB)^T = B^T A^T$ ，可推广到字符串翻转

$P_{camera} = P_{object} M_{object \rightarrow world} M_{world \rightarrow camera}$

线性变换： $F(\mathbf{a}+\mathbf{b}) = F(\mathbf{a})+F(\mathbf{b})$ ， $F(k\mathbf{a}) = kF(\mathbf{a})$ ，则称映射 F 是线性的
(\mathbf{aM} 满足此条件)

仿射变换：线性变换后接平移， $\mathbf{v}' = \mathbf{vM} + \mathbf{b}$

对 \mathbf{aM} ，求逆变换等价于求矩阵的逆

矩阵行列式： $|\mathbf{M}| = \sum_{j=1}^n m_{ij} c_{ij} = \sum_{j=1}^n m_{ij} (-1)^{i+j} |\mathbf{M}^{\{ij\}}|$

矩阵的逆： $M(M^{-1}) = M^{-1}M = I$ ，不可逆矩阵又称奇异矩阵，奇异矩阵行列式为0

标准伴随矩阵： $\text{adj}\mathbf{M}$ ， \mathbf{M} 的代数余子式矩阵的转置矩阵。 $M^{-1} = \frac{\text{adj}\mathbf{M}}{|\mathbf{M}|}$

正交矩阵： $MM^T = I$ ，旋转和镜像矩阵是正交矩阵。正交矩阵满足：矩阵的每一行都是单位向量，矩阵的所有行相互垂直。

Vector4, 齐次坐标。(x, y, z, w)实际代表3D中的(x/w, y/w, z/w)

旋转矩阵：描述一个坐标中基向量到另一个坐标基向量的转换。

矩阵蠕变：由于浮点数精度有限导致误差积累。

欧拉角：heading-pitch-bank约定。

万向锁：三个角度不互相独立，一旦选择正负90度为pitch角，就被限制在只能绕竖直轴旋转，失去了一个维度。

四元数：

几何图元自由度：是无歧义的描述该实体所需信息量的最小数目。

射线：是一个有向线段，参数形式： $x(t) = x_0 + t\Delta x$ $y(t) = y_0 + t\Delta y$ ，

向量记法： $p(t) = p_0 + td$ ，增量向量 d 指定了它的长度和方向，斜截式：

$y = mx + b$

球和圆：

Sublime Text 2

ctrl+shift+up/down: move line up/down

ctrl+alt+up/down: block edit up/down

Swift

<http://blackblake.synology.me/wordpress/?p=29>: Swift里的Optional和Unwrapping

PhotoShop

alt+ctrl+c: Resize Canvas

alt+ctrl+shift+s: Save for web

ctrl+h: show canvas guides; ctrl+mouse drag ruler: add canvas guide

LaTeX

%!Mode:: "TeX:UTF-8": make WinEdt show Chinese

Git

gitk file/folder: show commit with file

Windows

放大镜: ctrl+alt+d: 停靠模式; ctrl+alt+l: 窗口模式; win++: 放大;

win+esc: 退出放大镜

JavaScript

JavaScript组成: ECMAScript, DOM:针对XML文件的操作接口, BOM:浏览器对象模型, HTML5标准化

浮点数误差: $0.1 + 0.2 = 0.3000000000004$, 通过x10法解决