北京大学

Graphics & Interactive Lab

# 计算机图形学研究生课程班
# 图形变换


讲授：李　胜

北京大学信息科学技术学院

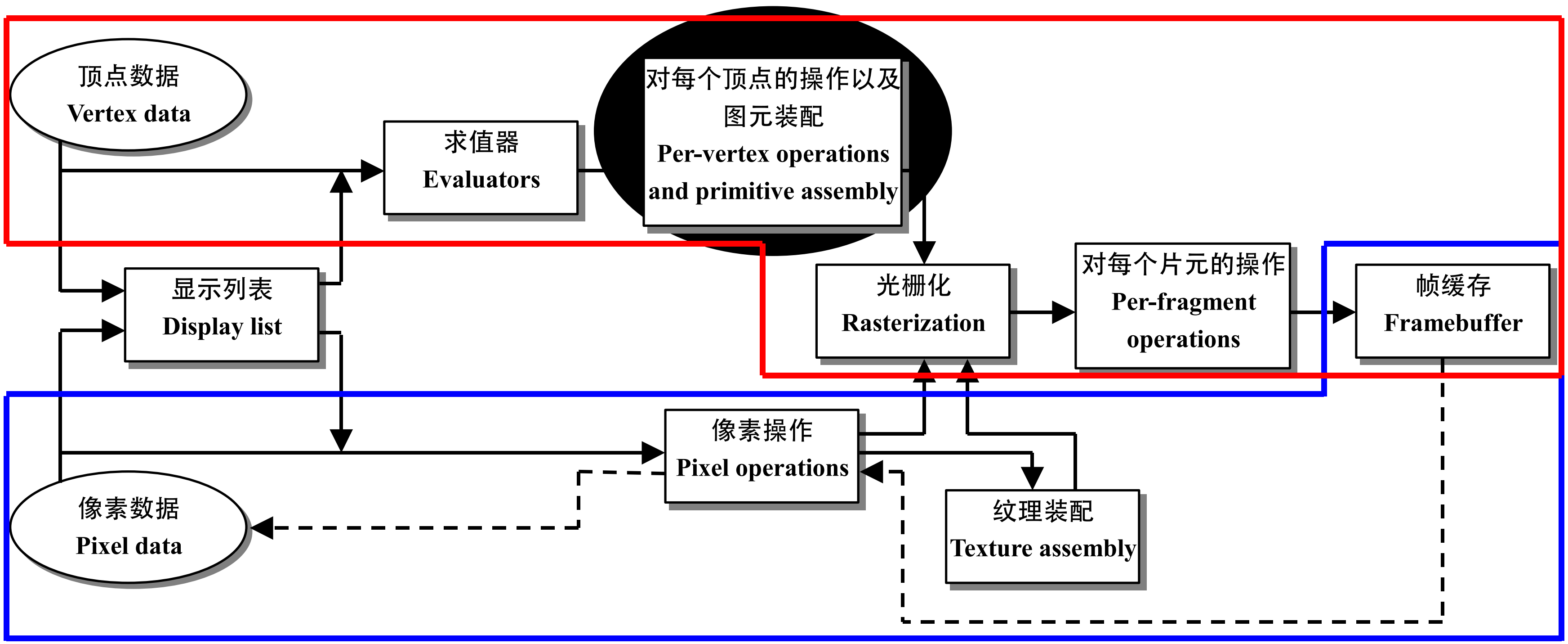
图形与交互技术实验室

实验室：理科1楼1323N，电话：13522401908

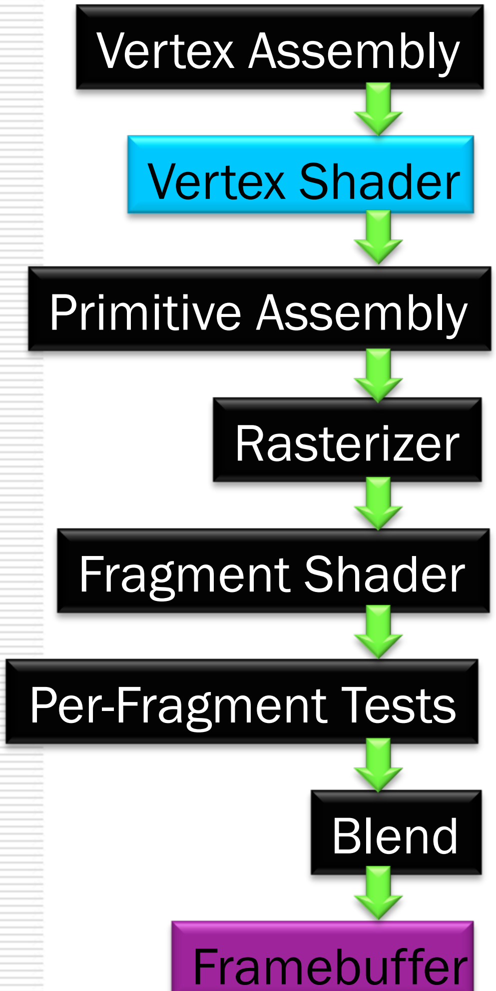
邮箱：lisheng@pku.edu.cn

# 回顾（一）

- OpenGL渲染流水线的两条数据处理路径
  - 顶点数据，基本的几何数据
  - 像素数据，图像特效，纹理数据

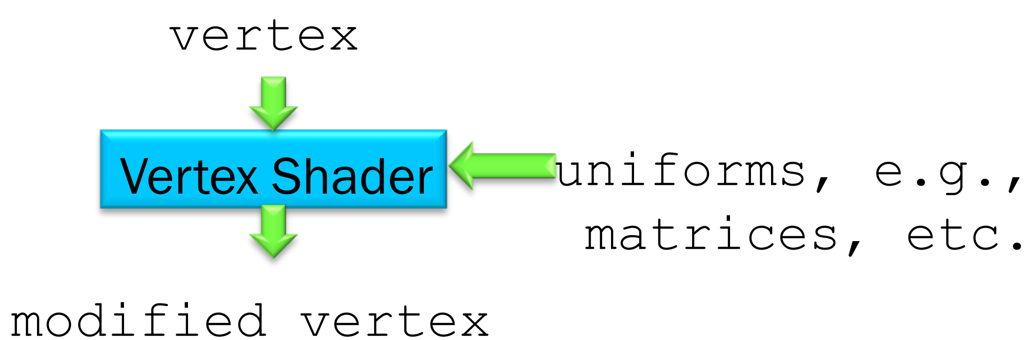


# 回顾（二）




- Model to clip coordinates requires three transforms:
  - model to world
  - world to eye
  - eye to clip
- Use 4x4 matrices passed to the vertex shader as uniforms

$$P_{world} = (M_{model})(P_{model})$$
$$P_{eye} = (M_{view})\ (P_{world})$$
$$P_{clip} = (M_{projection})\ (P_{eye})$$



# 提纲

- 变换的数学基础
- 二维基本变换
- 齐次坐标与二维变换的矩阵表示
- 复合变换与变换的模式
- 其它变换
- 二维图形的显示流程图
- 窗口到视区的变换
- 三维几何变换
- 坐标系之间的变换

# 变换的数学基础(1/4)

- 矢量

$$U = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \qquad V = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

- **矢量和**

$$U + V = \begin{bmatrix} u_x + v_x \\ u_y + v_y \\ u_z + v_z \end{bmatrix}$$

# 变换的数学基础(2/4)

- 矢量的数乘

$$k \bullet U = \begin{bmatrix} ku_x \\ ku_y \\ ku_z \end{bmatrix}$$

- 矢量的点积

$$U \bullet V = u_x v_x + u_y v_y + u_z v_z$$

  - 性质

$$U \bullet V = V \bullet U$$

$$U \bullet V = 0 \Leftrightarrow U \perp V$$

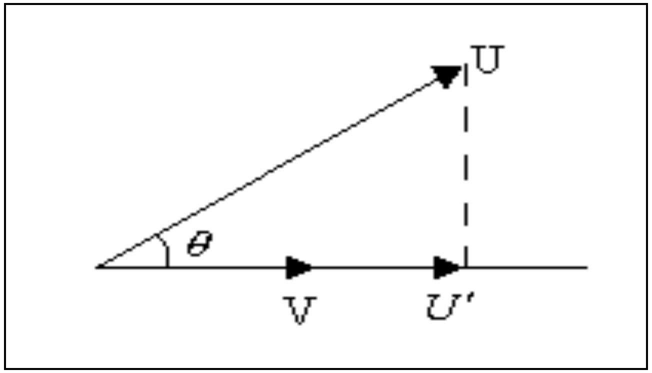
$$U \bullet U = 0 \Leftrightarrow U = 0$$

# 变换的数学基础(3/4)

- **矢量的长度**

$$\|U\| = \sqrt{U \bullet U} = \sqrt{u_x^{\;2} + u_y^2 + u_z^2}$$

  - 单位矢量
  - 点积运算的几何解释
  - 矢量的夹角




$$\cos\theta = \frac{U \bullet V}{\|U\| \bullet \|V\|}$$

- **矢量的叉积**

$$U \times V = \begin{vmatrix} i & j & k \\ u_x & u_y & u_z \\ v_x & v_y & v_z \end{vmatrix}$$

# 变换的数学基础(4/4)

- 矩阵
  - $m \times n$ 阶矩阵
  - n阶方阵
  - 零矩阵
  - 行向量与列向量
  - 单位矩阵 I
  - 矩阵的加法

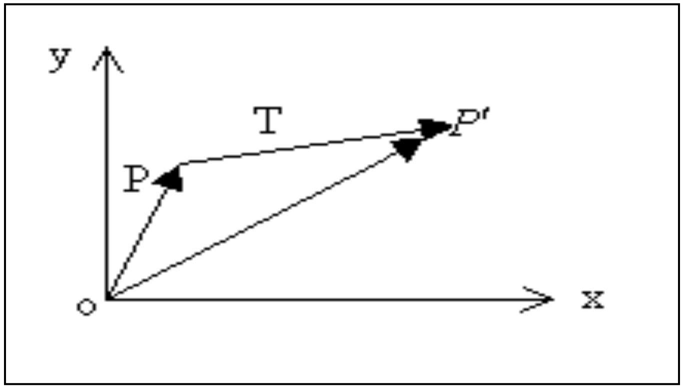
  结合律： ABC =A(BC) = (AB)C

  分配律：A(B+C) = AB+AC (B+C)A=BA+CA
  - 矩阵的数乘
  - 矩阵的乘法
  - 矩阵的转置
  - 矩阵的逆

# 二维基本变换（1/3）

- 平移变换

$$P' = P + T$$




$$P' = \begin{bmatrix} x' \\ y' \end{bmatrix} \qquad P = \begin{bmatrix} x \\ y \end{bmatrix} \qquad T = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

$$\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases}$$

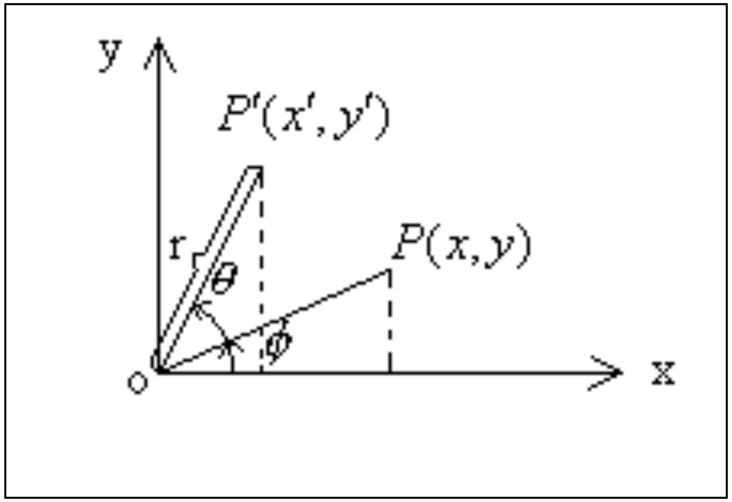
# 二维基本变换（2/3）

## 旋转变换

- 绕坐标原点旋转角度 $\theta$ （逆时针为正，顺时针为负）

$$P' = R \bullet P$$

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

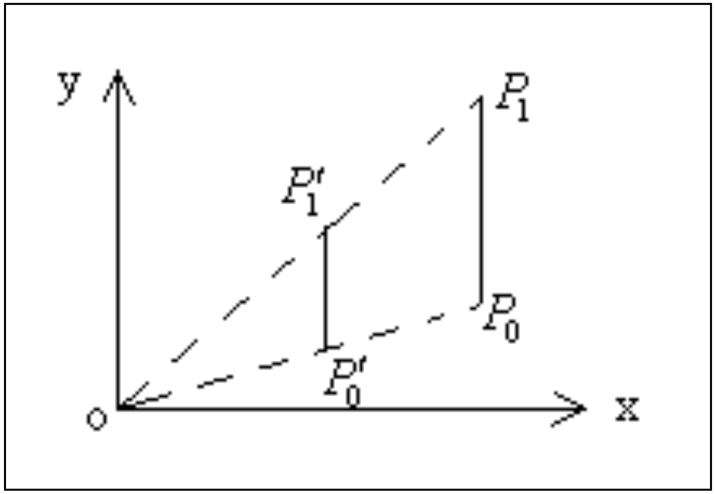


# 二维基本变换（3/3）

- 放缩变换

$$P' = S \bullet P$$

$$S = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$$




- **以坐标原点为放缩参照点**
- **不仅改变了物体的大小和形状，也改变了它离原点的距离**

# 齐次坐标与二维变换的矩阵表示(1/4)

- 为什么需要齐次坐标？

多个变换作用于多个目标

变换合成

变换合成的问题

引入齐次坐标

变换的表示法统一

# 齐次坐标与二维变换的矩阵表示(2/4)

## 齐次坐标

- **定义** ：所谓齐次坐标就是将一个原本是n维的向量用一个n+1维向量来表示 $(x_h, y_h, h)$
  - (x,y)点对应的齐次坐标为

$$x_h = hx, y_h = hy, h \neq 0$$

- (x,y)点对应的齐次坐标为三维空间的一条直线

$$\begin{cases} x_h = hx \\ y_h = hy \\ z_h = h \end{cases}$$

# 齐次坐标与二维变换的矩阵表示(3/4)

- 标准齐次坐标(x,y,1)
- 二维变换的矩阵表示
  - **平移变换**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \overset{\text{记为}}{=} T(t_x, t_y) \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

  - **旋转变换**

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \overset{\text{记为}}{=} R(\theta) \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

# 齐次坐标与二维变换的矩阵表示(4/4)

- 放缩变换

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \overset{\text{记为}}{=} S(s_x, s_y) \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

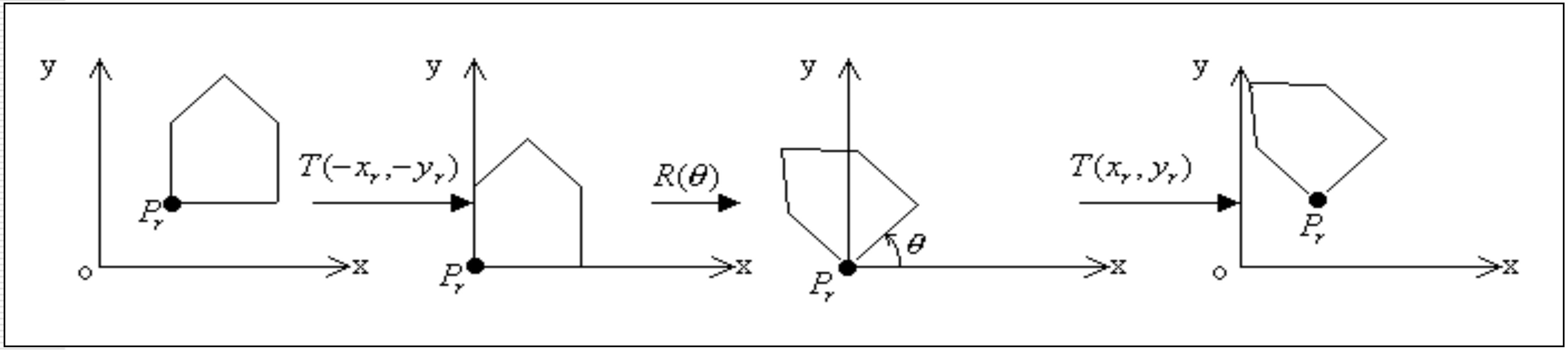
- 变换具有统一表示形式的优点
  - 便于变换合成
  - 便于硬件实现

# 复合变换及变换的模式（1/6）

- 问题：如何实现复杂变换？

变换分解 ⟶ 变换合成

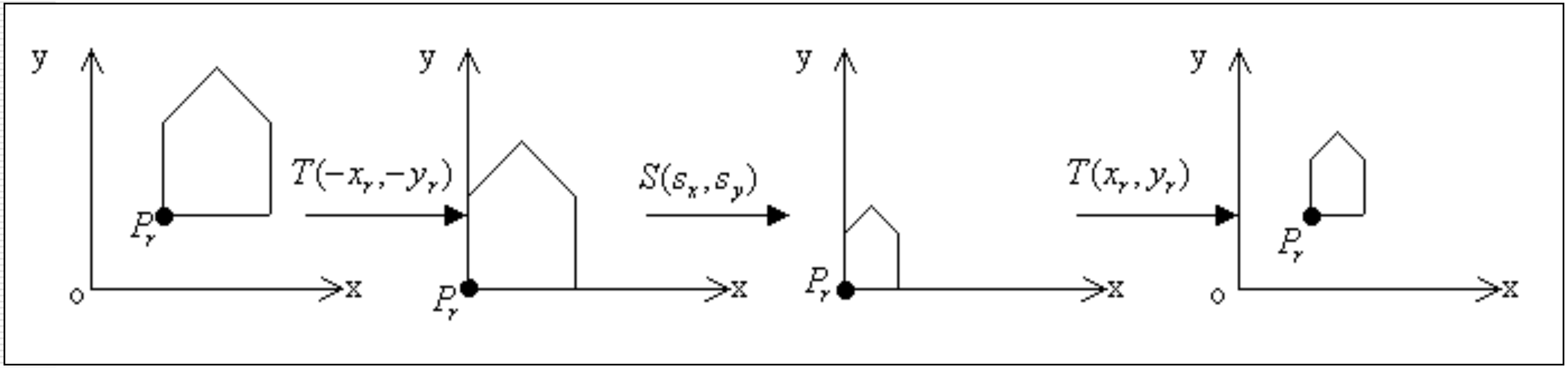
- 关于任意参照点 $P_r(x_r, y_r)$ 的旋转变换




$$R(x_r, y_r; \theta) = T(x_r, y_r) \bullet R(\theta) \bullet T(-x_r, -y_r)$$

# 复合变换及变换的模式（2/6）

- 关于任意参照点 $P_r(x_r, y_r)$ 的放缩变换



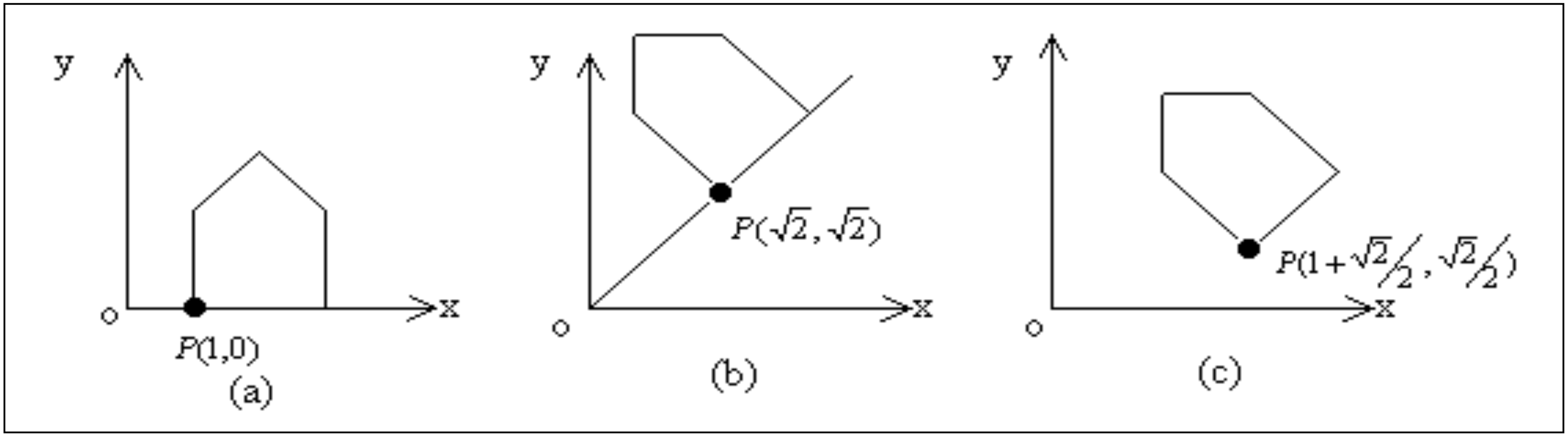

$$S(x_r, y_r; s_x, s_y) = T(x_r, y_r) \bullet S(s_x, s_y) \bullet T(-x_r, -y_r)$$

# 复合变换及变换的模式（3/6）

- 变换的结果与变换的顺序有关（矩阵乘法不可交换）

*Translate2D(1,0);*
*Rotate2D(45);*
*House();*

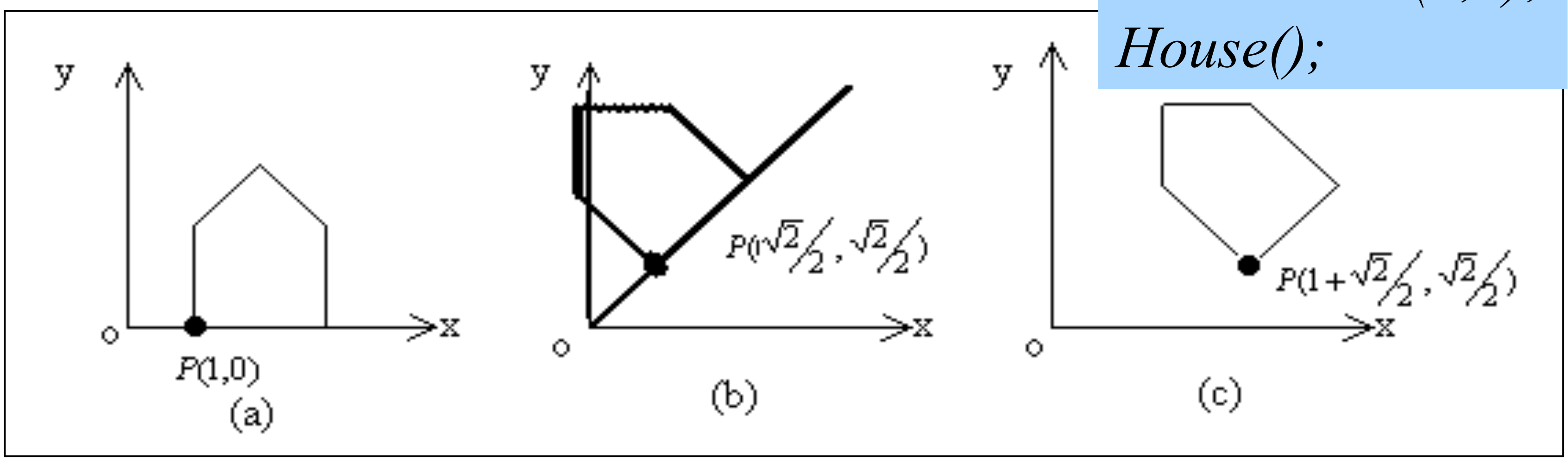
*Rotate2D(45);*
*Translate2D(1,0);*
*House();*



# 复合变换及变换的模式（4/6）

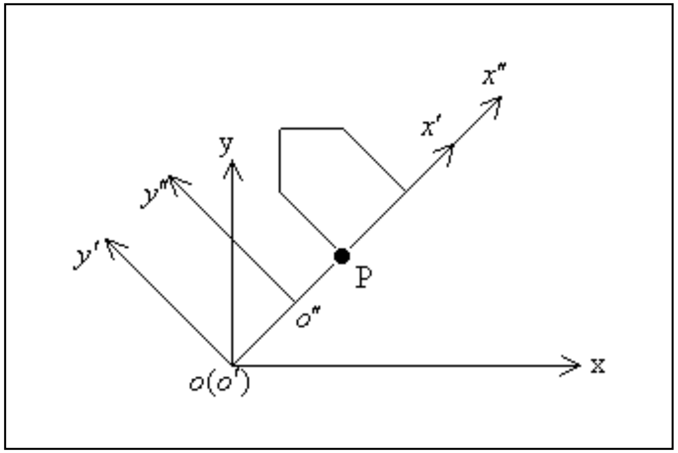
- **变换的固定坐标系模式**
  - 相对于同一个固定坐标系
  - 先调用的变换先执行，后调用的变换后执行

*Rotate2D(45);*
*Translate2D(1,0);*
*House();*



# 复合变换及变换的模式（5/6）

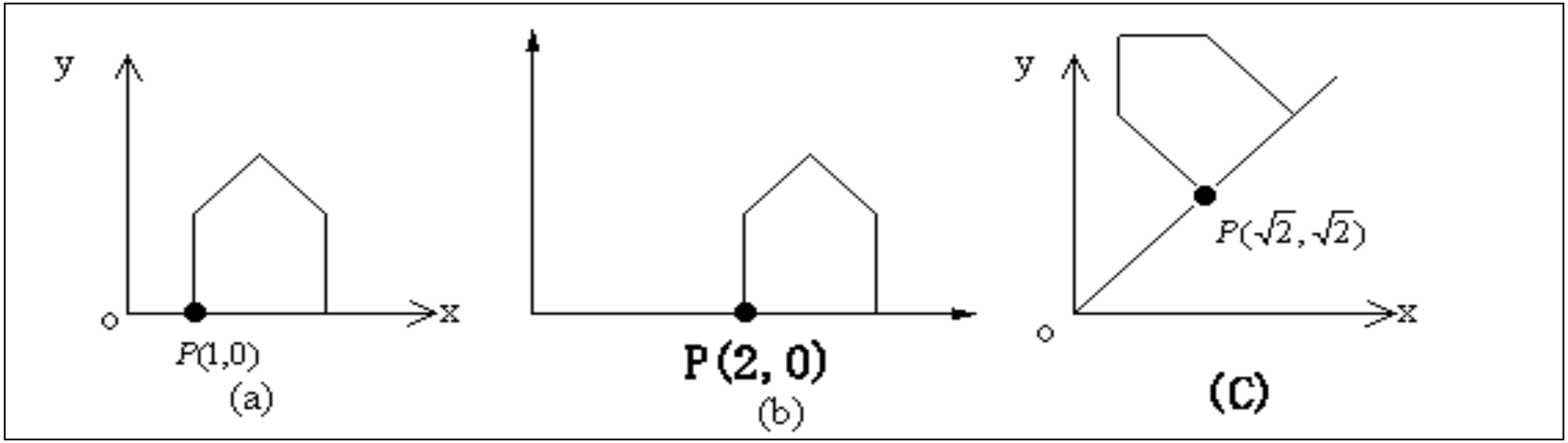
- 人的思维方式
  - 每次变换产生一个新的坐标系




- 变换的活动坐标系模式
  - 先调用的变换后执行，后调用的变换先执行（图形系统一般用堆栈实现）

# 复合变换及变换的模式（6/6）

*Rotate2D(45);*
*Translate2D(1,0);*
*House();*

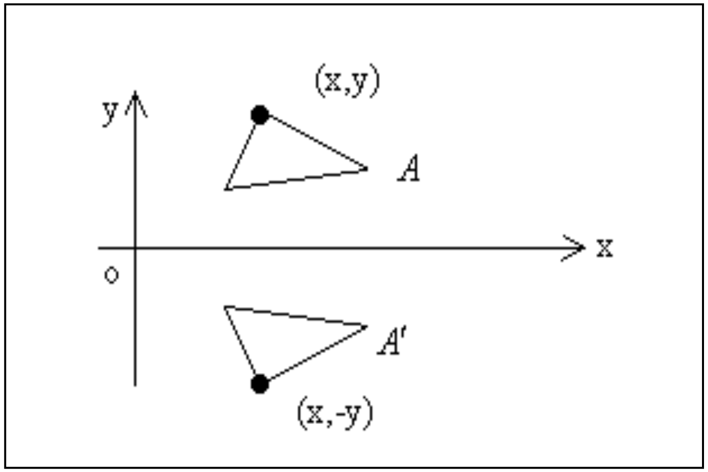



例子

# 其它变换（1/6）

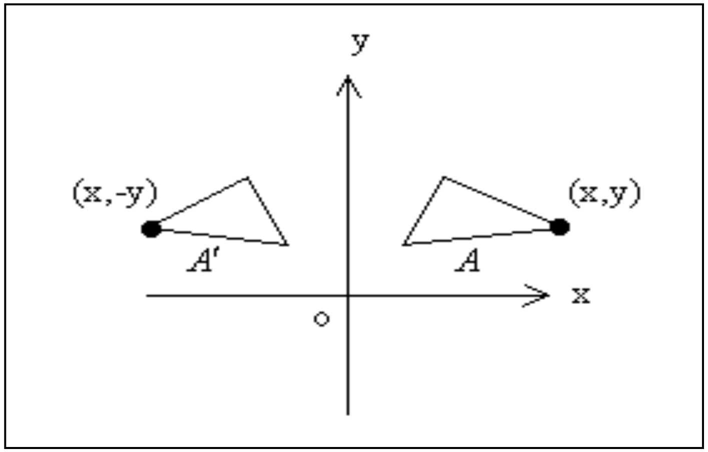
- 对称变换
  - 关于x轴的对称变换

$$SY_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



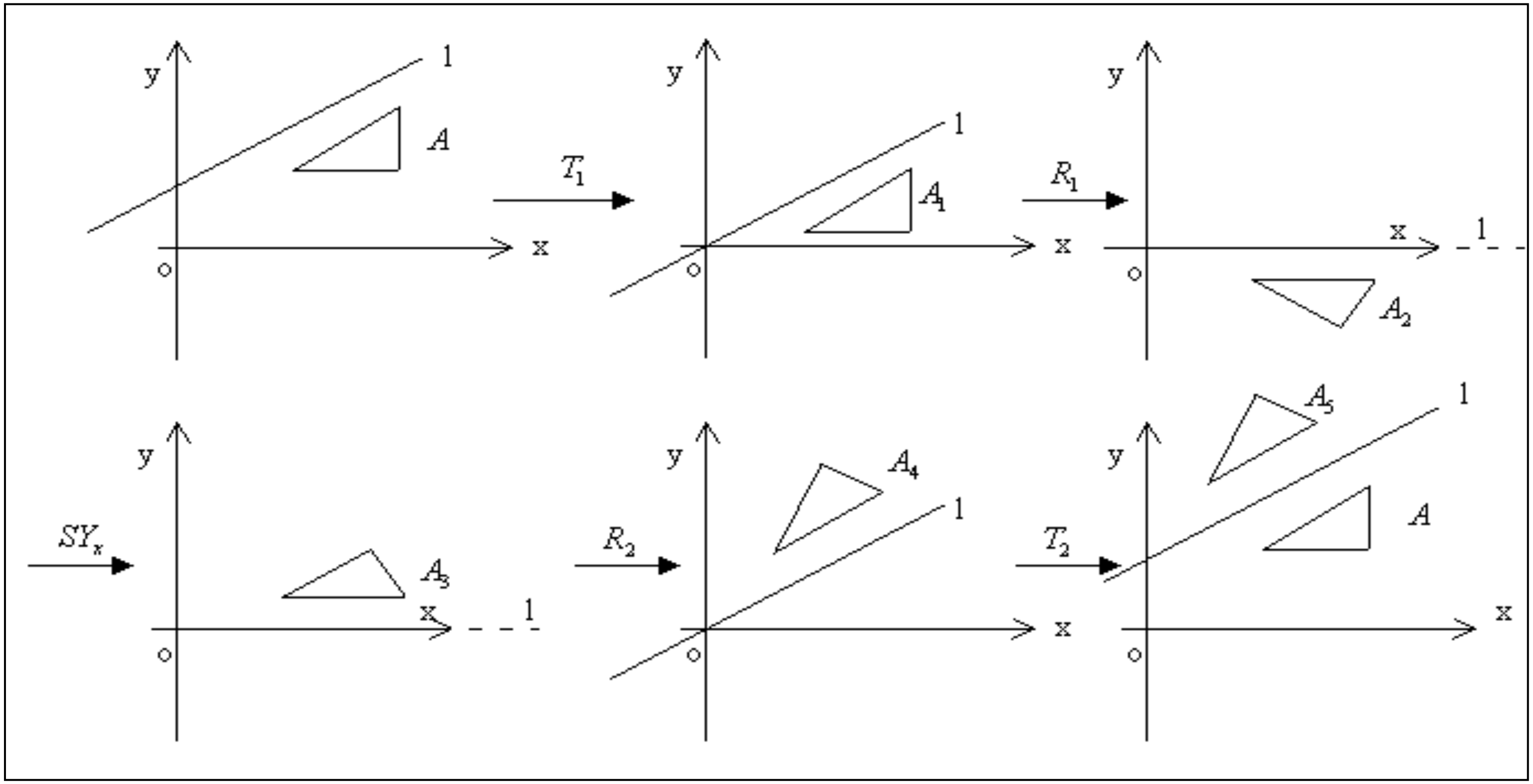

  - 关于y轴的对称变换

$$SY_y = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



# 其它变换（2/6）

- 关于任意轴的对称变换



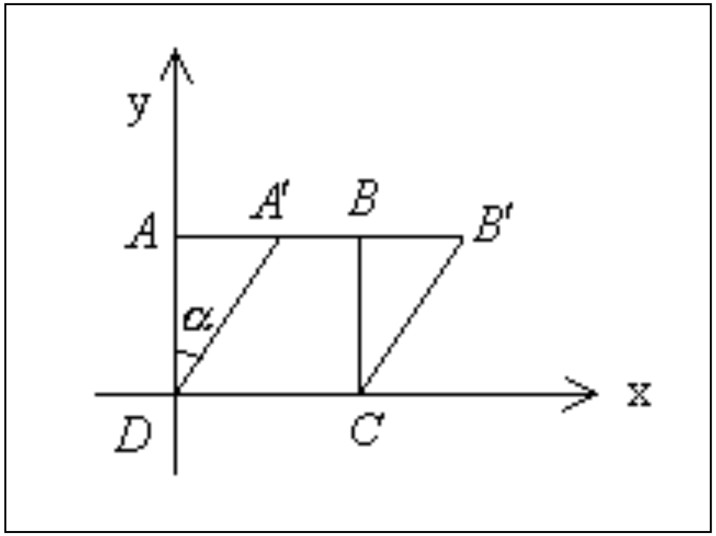
# 其它变换（3/6）

## 错切变换

- **以y轴为依赖轴的错切变换**
  - 以y=0为参考轴

$$\begin{cases} x' = x + sh_x y \\ y' = y \end{cases}$$

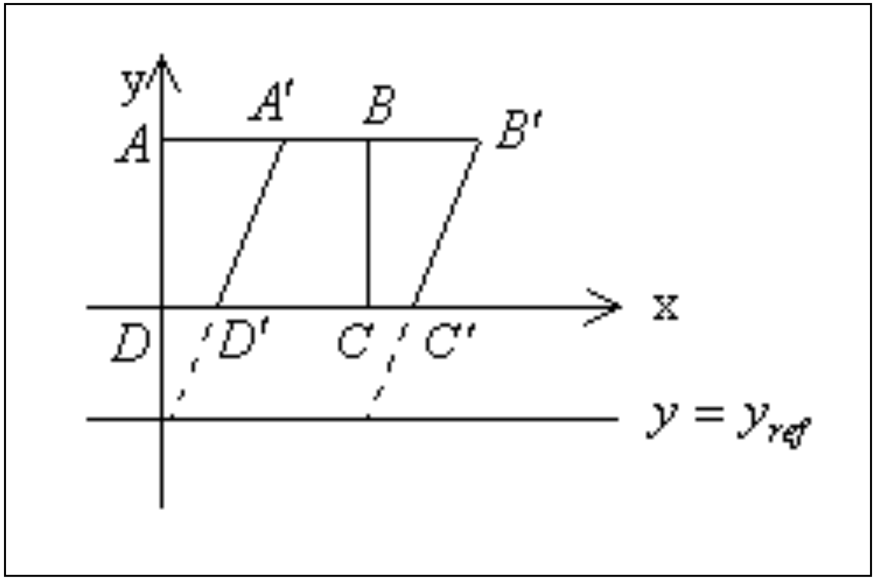
$$SH_y(sh_x) = \begin{bmatrix} 1 & sh_x & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$




  - $sh_x = \tan a$

# 其它变换（4/6）

- 以 $y = y_{ref}$ 为参考轴



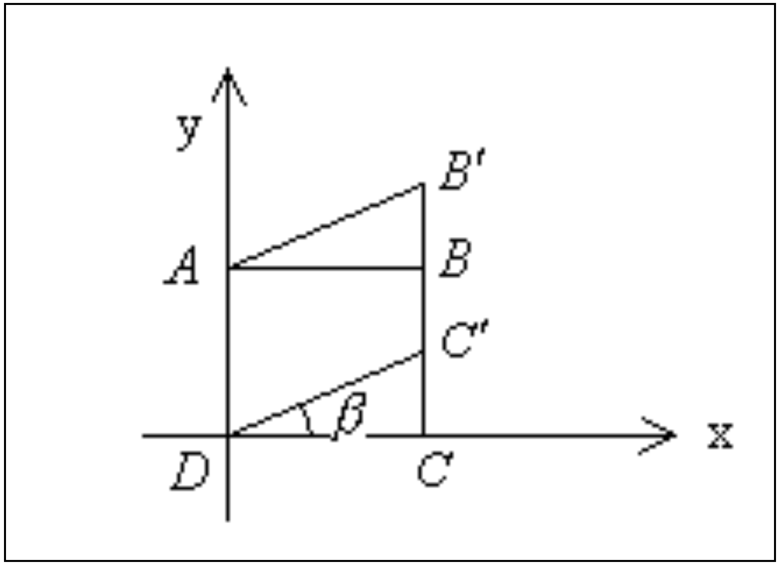

$$SH_y(sh_x, y_{ref}) = \begin{bmatrix} 1 & sh_x & -sh_x \bullet y_{ref} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

# 其它变换（5/6）

- 以x轴为依赖轴的错切变换

$$\begin{cases} x' = x \\ y' = sh_y x + y \end{cases}$$

$$SH_x(sh_y) = \begin{bmatrix} 1 & 0 & 0 \\ sh_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



# 其它变换（6/6）

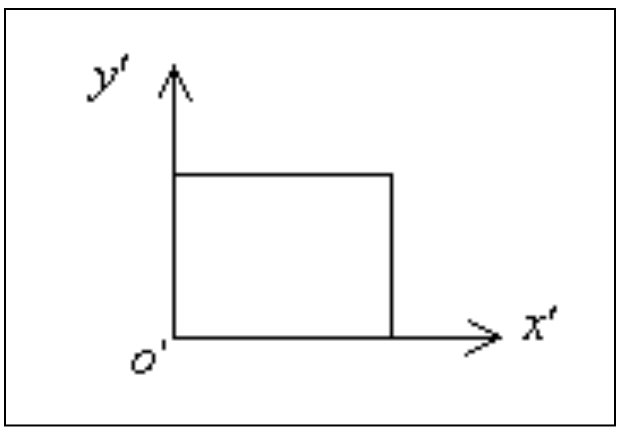
- 仿射变换
- 几何变换的一般形式，保持平行直线的平行性

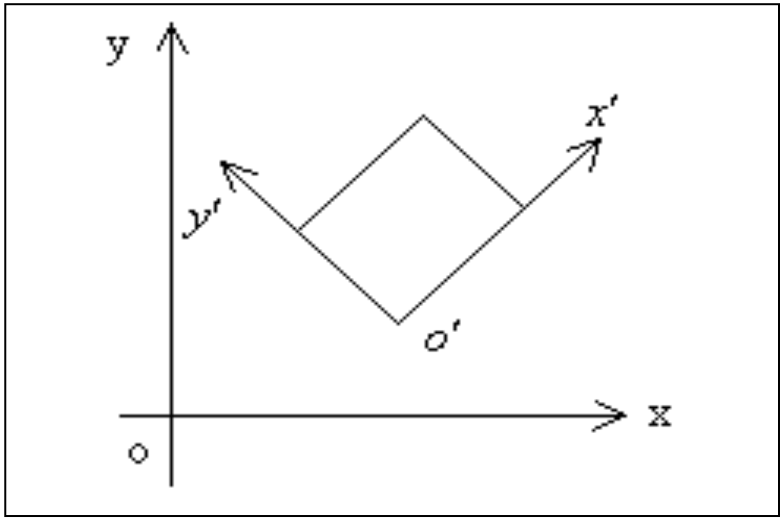
$$Af = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix}$$

# 二维图形的显示流程图（1/4）

- 坐标系：建立了图形与数之间的联系
  - **世界坐标系(world coordinate)**

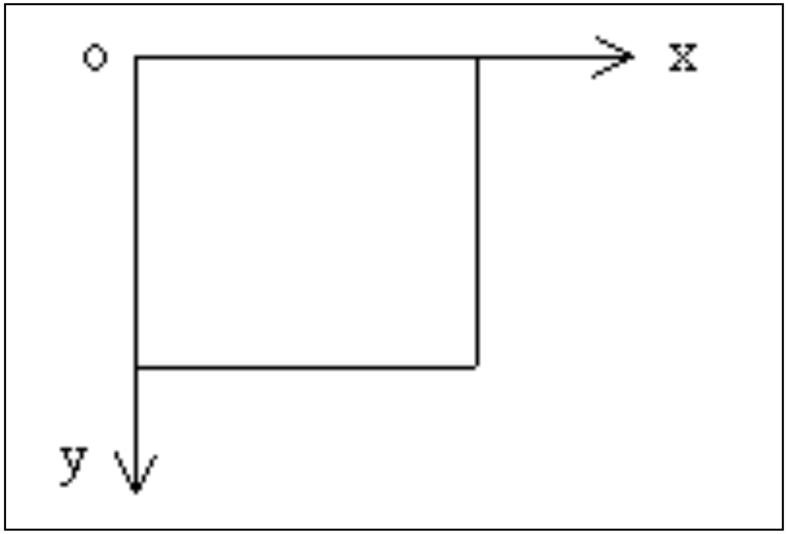
    **或者 用户坐标系(user coordinate)**
  - **局部坐标系(local coordinate)**



# 二维图形的显示流程图（2/4）

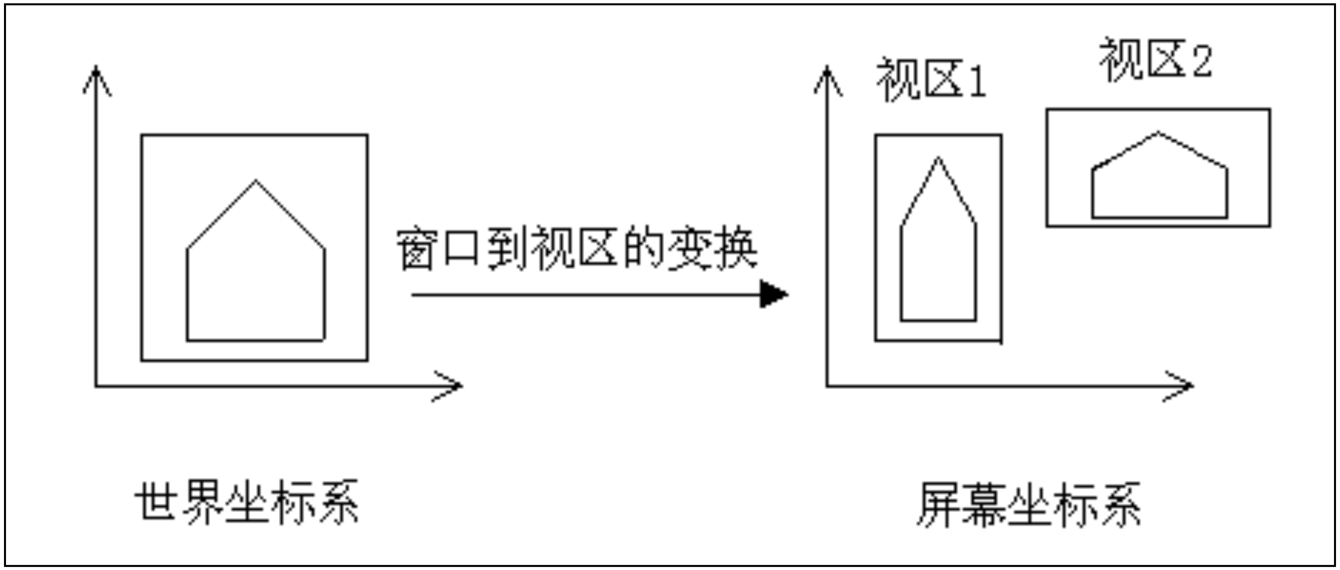
- 屏幕坐标系(screen coordinate)
  或者  设备坐标系(device coordinate)

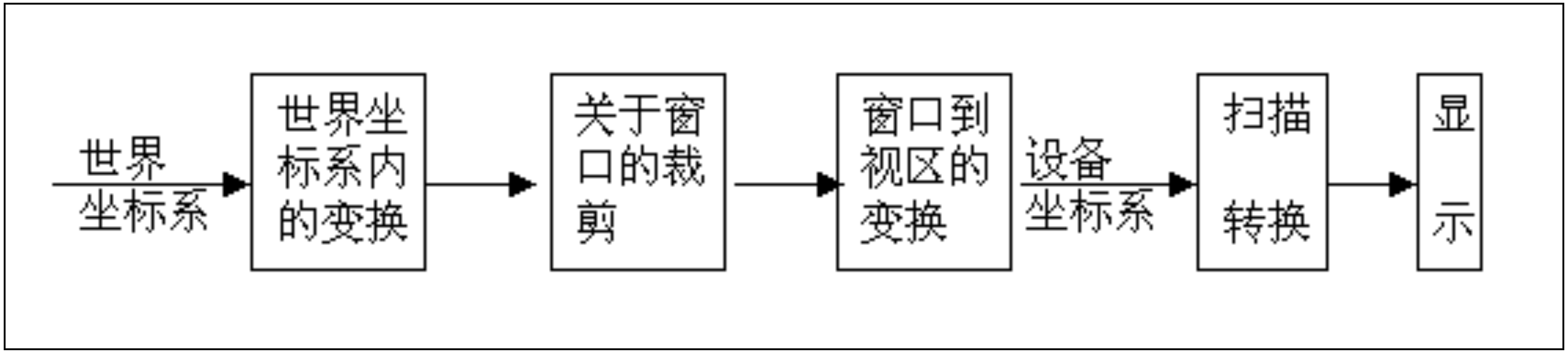


# 二维图形的显示流程图（3/4）

- 窗口
  - 在世界坐标系中指定的矩形区域
  - 用来指定要显示的图形
- 视区
  - 在设备坐标系（屏幕或绘图纸）上指定的矩形区域
  - 用来指定窗口内的图形在屏幕上显示的大小及位置
- 窗口到视区的变换

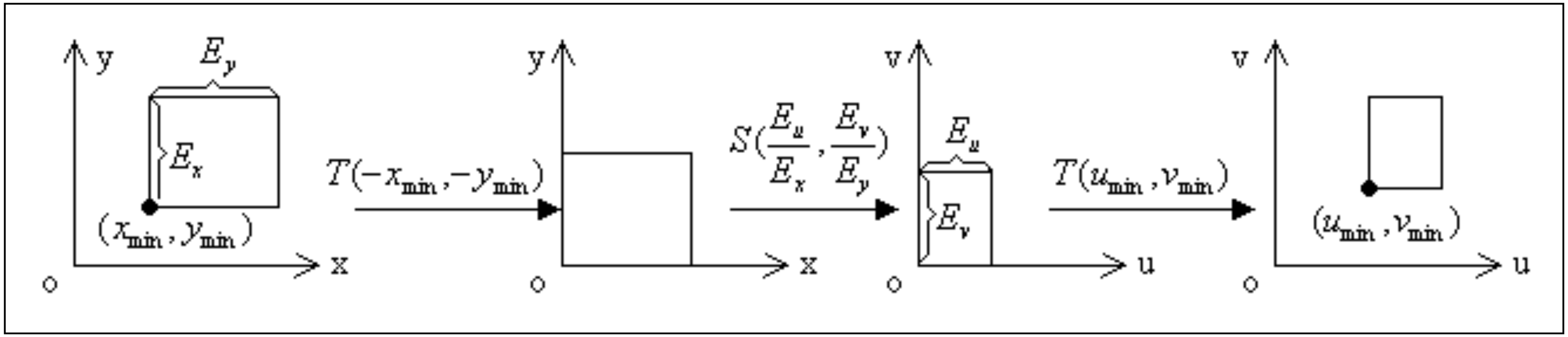
# 二维图形的显示流程图（4/4）



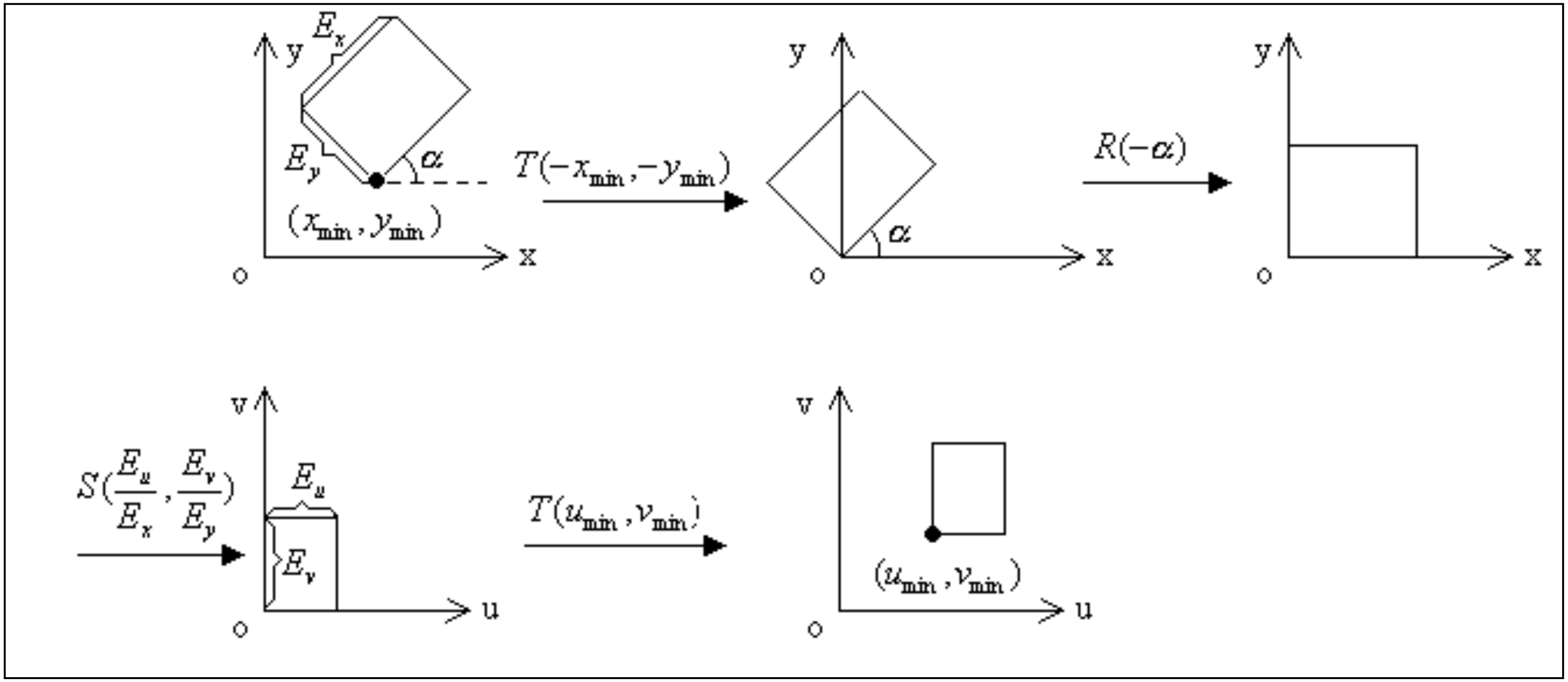
# 窗口到视区的变换（1/2）

- 目标
  - 将窗口之中的图形变换到视区中
- 变换的求法
  - 变换的分解与合成




$$M_{wv} = T(u_{\min}, v_{\min}) S(\frac{E_x}{E_x}, \frac{E_v}{E_y}) T(-x_{\min}, -y_{\min})$$

# 窗口到视区的变换（2/2）




$$M_{wv} = T(u_{min}, v_{min}) S(\frac{E_x}{E_x}, \frac{E_v}{E_y}) R(-\alpha) T(-x_{min}, -y_{min})$$

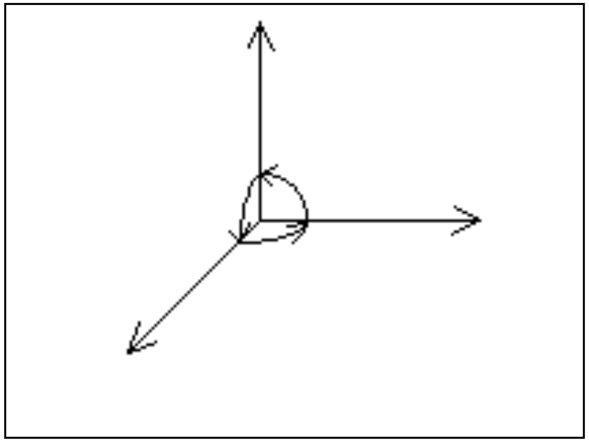
# 三维几何变换（1/5）

- 三维齐次坐标
  - (x,y,z)点对应的齐次坐标为

$$(x_h, y_h, z_h, h)$$

$$x_h = hx, y_h = hy, z_h = hz, h \neq 0$$

  - 标准齐次坐标(x,y,z,1)
- 右手坐标系



# 三维几何变换（2/5）

- 平移变换

$$T(t_x, t_y, t_z) = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- 放缩变换

$$S(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# 三维几何变换（3/5）

## 旋转变换

- **绕x轴**

$$R_x(\theta)=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\theta & -\sin\theta & 0\\ 0 & \sin\theta & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

- **绕y轴**

$$R_y(\theta)=\begin{bmatrix}\cos\theta & 0 & \sin\theta & 0\\ 0 & 1 & 0 & 0\\ -\sin\theta & & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

# 三维几何变换（4/5）

- 绕z轴

$$R_z(\theta)=\begin{bmatrix}\cos\theta & -\sin\theta & 0 & 0\\ \sin\theta & \cos\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

- 错切变换

$$SH_z(sh_x,sh_y)=\begin{bmatrix}1 & 0 & sh_x & 0\\ 0 & 1 & sh_y & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

# 三维几何变换（5/5）

- 对称变换
  - 关于坐标平面xy的对称变换

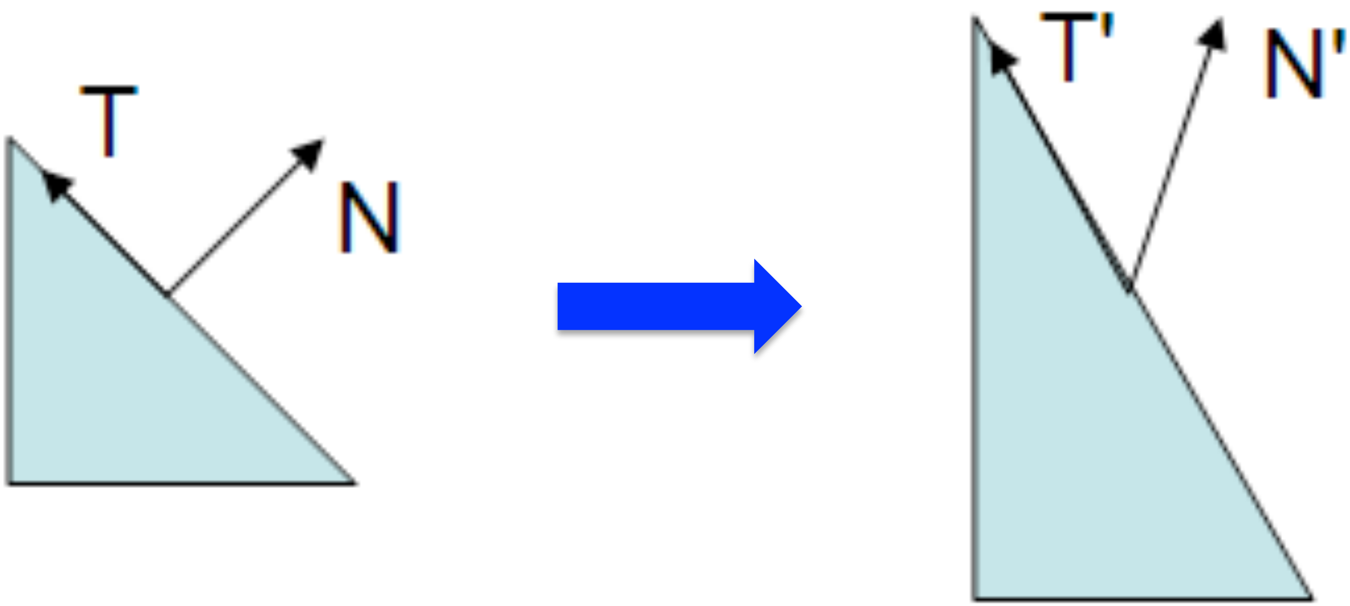
$$SY_{xy} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- 三维变换的一般形式

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# 法向如何变换？

直接使用模型变换矩阵？



# 法向如何变换？

法向的变换矩阵求解：

$$N' \cdot T' = (GN) \cdot (MT) = 0$$

$$(GN) \cdot (MT) = (GN)^T (MT)$$

$$(GN)^T (MT) = N^T G^T M T$$

满足上式为0的条件： $G^T M = I$

$$G = (M^{-1})^T$$

# 坐标系之间的变换

- 什么是？
  - 建立坐标系之间的变换关系
  - 将图形从一个坐标系中变换到另一个坐标系中
- 怎样求？

（思考。。。）