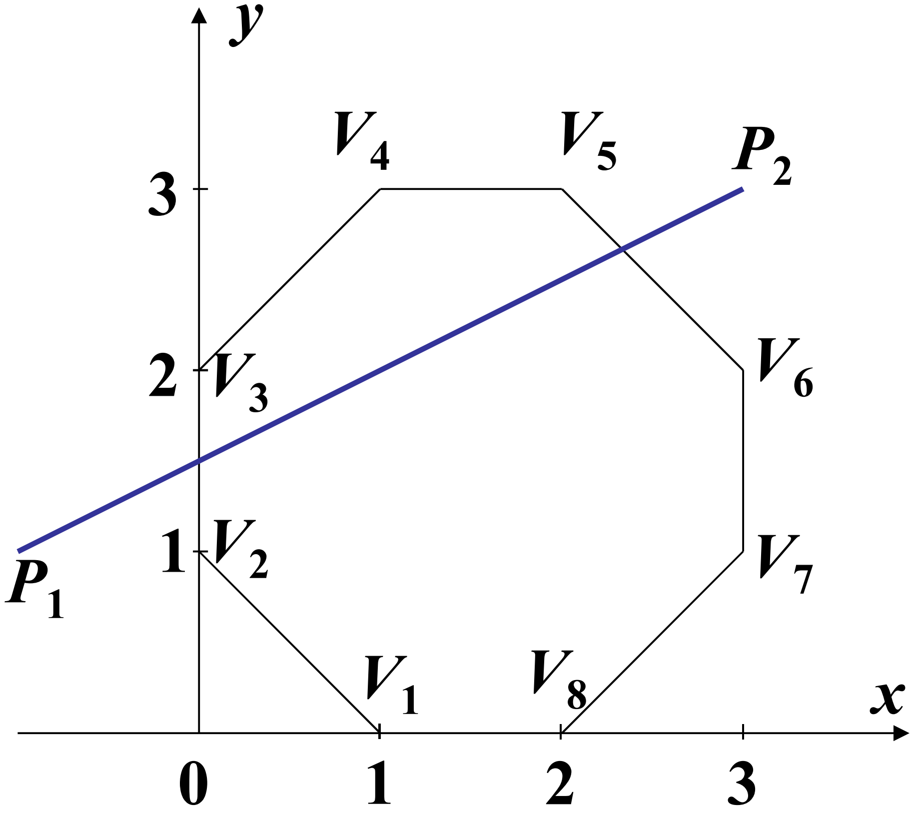
上节要点

- 线段的扫描转换(**DDA**/中点/**Bresenham**)
- 圆弧的扫描转换(中点/正负/多边形逼近)
- 多边形的扫描转换(逐点判断/扫描线/边缘填充/边界标志)
- 多边形的区域填充(递归/扫描线)

上节要点

- 直线段的裁剪
  - 直接求交/**Cohen-Sutherland**法/中点分割
  - **Nicholl-Lee-Nicholl**
  - 梁友栋-**Barsky**算法/ **Lyrus-Beck**算法
- 多边形的裁剪
  - **Sutherland-Hodgeman**算法
  - **Weiler-Atherton**算法



P_1(-1,1), P_2(3,3)

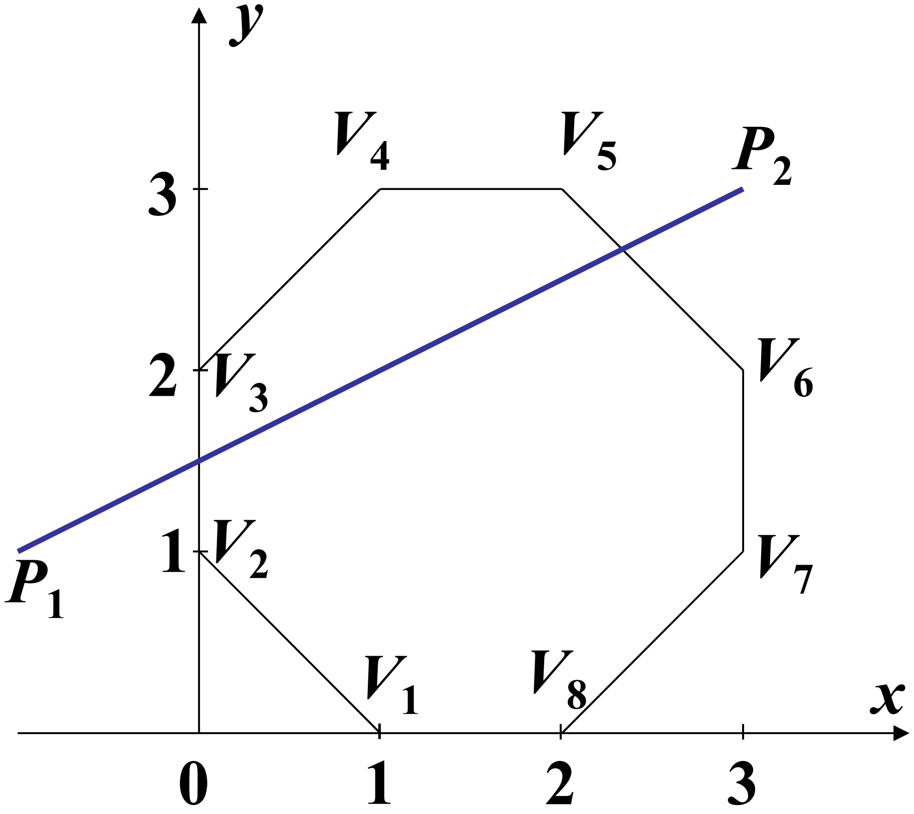
$$P(t)=\begin{bmatrix}-1\\1\end{bmatrix}+\begin{bmatrix}4\\2\end{bmatrix}t$$

当$N_i\cdot(P_2-P_1)>0$时，

$$t\geq-\frac{N_i\cdot(P_1-A_i)}{N_i\cdot(P_2-P_1)}$$

当$N_i\cdot(P_2-P_1)<0$时，

$$t\leq-\frac{N_i\cdot(P_1-A_i)}{N_i\cdot(P_2-P_1)}$$

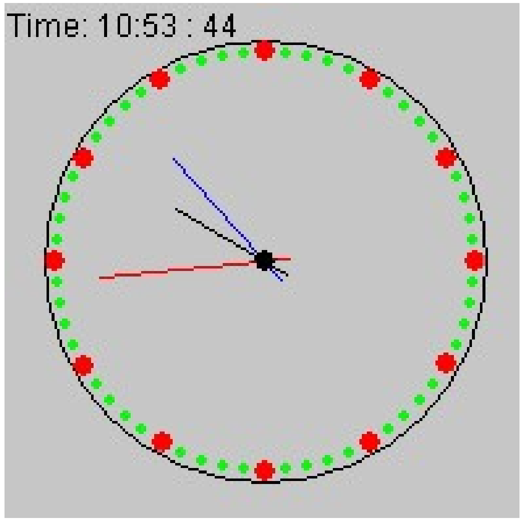


$P_1(-1,1)$, $P_2(3,3)$

$$P(t) = \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \begin{bmatrix} 4 \\ 2 \end{bmatrix} t$$

	N_i	A	$N_i \cdot (P_2-P_1)$	t_i
V_1V_2				
V_2V_3				
V_3V_4				
V_4V_5				
V_5V_6				
V_6V_7				
V_7V_8				
V_8V_1				

第五章　图形变换与投影



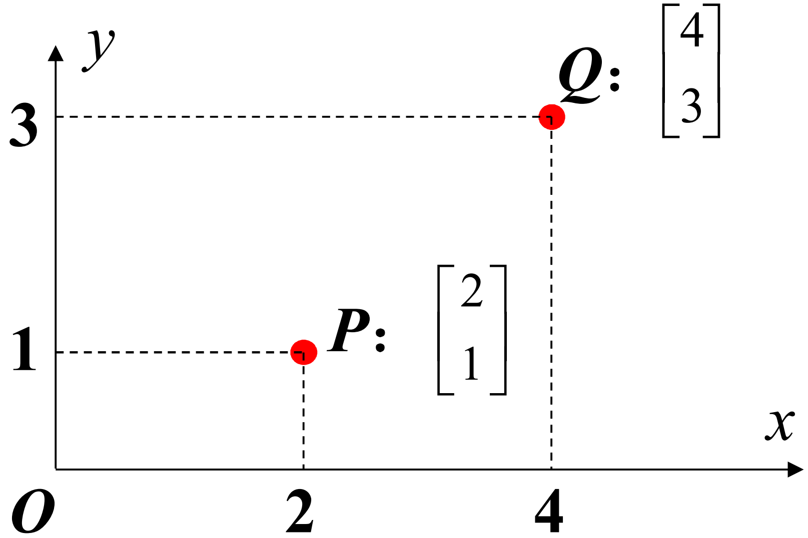
清华大学软件学院　张慧　2019-9-24

本章内容

- 图形变换的数学基础
- 齐次坐标与二维图形的几何变换
- 二维图形的显示流程
- 三维图形的几何变换
- 投影变换
- 三维图形的显示流程

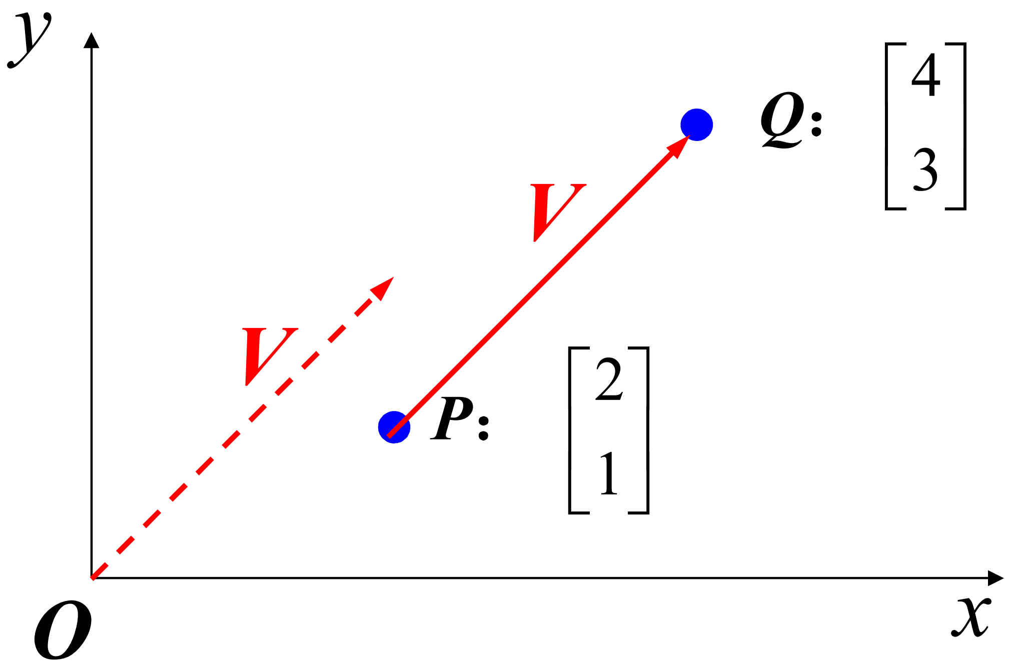
图形变换的数学基础

- 点



二维点 $P=\begin{bmatrix} p_x \\ p_y \end{bmatrix}$ 三维点 $P=\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$ **n**维点 $P=\begin{bmatrix} p_0 \\ p_1 \\ \vdots \\ p_{n-1} \end{bmatrix}$

■ 向量/矢量

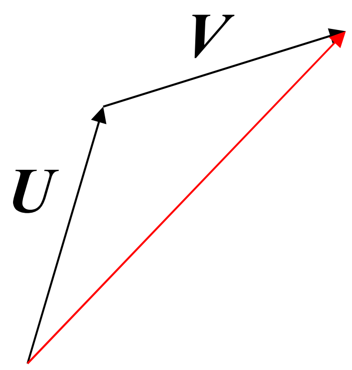


$$V = Q - P = \begin{bmatrix} q_x - p_x \\ q_y - p_y \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

- ***n*维向量**

$$V = \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_{\text{n-1}} \end{bmatrix} = \begin{bmatrix} q_0 - p_0 \\ q_1 - p_1 \\ \vdots \\ q_{\text{n-1}} - p_{n-1} \end{bmatrix}$$

$$U = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{\text{n-1}} \end{bmatrix}, \quad V = \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_{\text{n-1}} \end{bmatrix}$$



- **向量和**

$$U + V = \begin{bmatrix} u_0 + v_0 \\ u_1 + v_1 \\ \vdots \\ u_{\text{n-1}} + v_{n-1} \end{bmatrix}$$

- 向量的数乘

$$k \cdot V = \begin{bmatrix} kv_0 \\ kv_1 \\ \vdots \\ kv_{\text{n-1}} \end{bmatrix}$$

- 向量的点积

$$U \cdot V = u_0 v_0 + u_1 v_1 \cdots u_{n-1} v_{n-1} = \sum_{i=0}^{n-1} u_i v_i$$

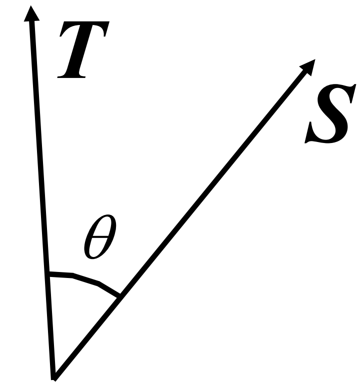
$$U \cdot V = \left|U\right| \cdot \left|V\right| \cos\theta$$

$$U \cdot V = V \cdot U$$

$$U \cdot U = 0 \Leftrightarrow U = 0$$

- 向量的长度： $|V| = \sqrt{VV} = \sqrt{V^2} = \sqrt{\sum_{i=0}^{n-1} v_i^{\ 2}}$
- 两向量间的夹角

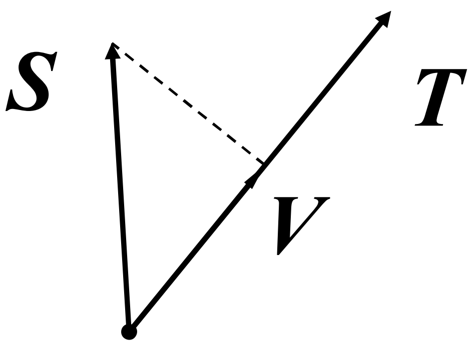
$$\cos\theta = \frac{S \cdot T}{|S| \cdot |T|}$$



- 两非零向量互相垂直充要条件

$$S \cdot T = 0 \Leftrightarrow S \perp T$$

- 向量***S***在向量***T***上的投影向量***V***



$$V = \left(S\frac{T}{|T|}\right)\frac{T}{|T|}$$

- 向量的叉积
  - 二维向量的叉积

$$U=\begin{bmatrix}u_x\\u_y\end{bmatrix},\quad V=\begin{bmatrix}v_x\\v_y\end{bmatrix},\quad U\times V=u_xv_y-u_yv_x$$

  - 三维向量的叉积

$$U=\begin{bmatrix}u_x\\u_y\\u_z\end{bmatrix},\quad V=\begin{bmatrix}v_x\\v_y\\v_z\end{bmatrix},\quad U\times V=\begin{bmatrix}u_yv_z-u_zv_y\\u_zv_x-u_xv_z\\u_xv_y-u_yv_x\end{bmatrix}$$

- 矩阵

$$A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

$\boldsymbol{A}$，$\boldsymbol{A_{m\times n}}$，$\boldsymbol{(a_{i,j})_{m\times n}}$

  - 零矩阵：$\boldsymbol{a_{i,j}=0}$
  - 方阵：$\boldsymbol{m=n}$
  - 行矩阵/行向量、列矩阵/列向量
  - 矩阵相等

■ 矩阵的加法

$$A+B=\begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & \cdots & a_{1n}+b_{1n} \\ a_{21}+b_{21} & a_{22}+b_{22} & \cdots & a_{2n}+b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}+b_{m1} & a_{m2}+b_{m2} & \cdots & a_{mn}+b_{mn} \end{bmatrix}$$

$$A+B=B+A$$

$$A+B+C=(A+B)+C=A+(B+C)$$

$$A+0=A$$

■ 矩阵的数乘

$$k \cdot A = \begin{bmatrix} ka_{11} & ka_{12} & \cdots & ka_{1n} \\ ka_{21} & ka_{22} & \cdots & ka_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ ka_{m1} & ka_{m2} & \cdots & ka_{mn} \end{bmatrix}$$

$$k(A+B) = kA + kB$$

$$k(A \cdot B) = (k \cdot A) \cdot B = A \cdot (k \cdot B)$$

$$(\alpha + \beta)A = \alpha A + \beta A$$

$$\alpha(\beta A) = (\alpha\beta)A$$

- 矩阵的乘法

$$C = (c_{i,j})_{m\times p} = A_{m\times n} \cdot B_{n\times p}$$

$$c_{i,j} = \sum_{l=1}^{n} a_{i,l} \times b_{l,j}$$

$$ABC = (AB)C = A(BC)$$

$$A(B+C) = AB + AC$$

$$(B+C)A = BA + CA$$

- 矩阵的转置

$$A^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}$$

$$\left(A^{\mathrm{T}}\right)^{\mathrm{T}} = A$$

$$(A+B)^{\mathrm{T}} = A^{\mathrm{T}} + B^{\mathrm{T}}$$

$$(kA)^{\mathrm{T}} = kA^{\mathrm{T}}$$

$$(A \cdot B)^{\mathrm{T}} = B^{\mathrm{T}} \cdot A^{\mathrm{T}}$$

- 单位矩阵

$$I = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

$$A_{m,n} \cdot I_n = A_{m,n} \qquad I_m A_{m,n} = A_{m,n}$$

- 矩阵的逆

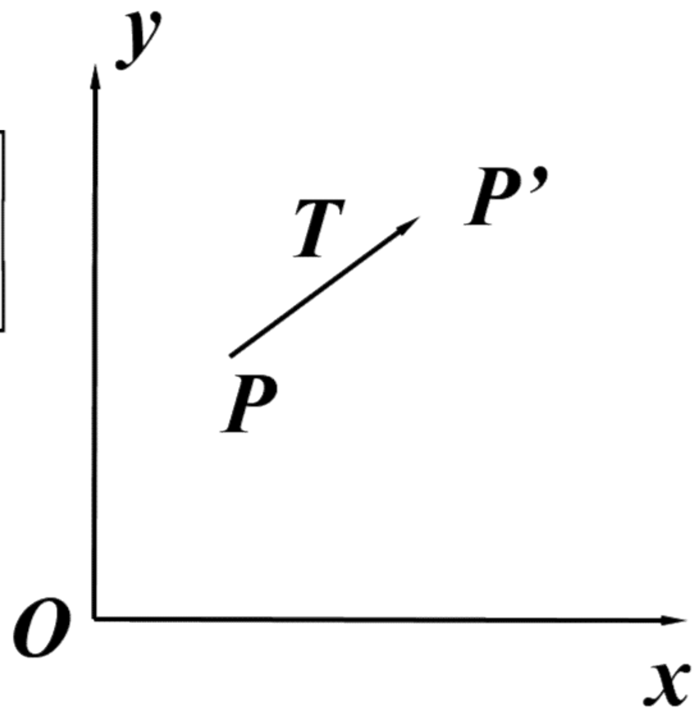
$$A \cdot B = B \cdot A = I$$

$$B = A^{-1}$$

二维图形的基本几何变换

- 平移变换

$$P=\begin{bmatrix} p_x \\ p_y \end{bmatrix} \qquad P'=\begin{bmatrix} p_x' \\ p_y' \end{bmatrix} \qquad T=\begin{bmatrix} t_x \\ t_y \end{bmatrix}$$



$$\begin{cases} p_x' = p_x + t_x \\ p_y' = p_y + t_y \end{cases}$$

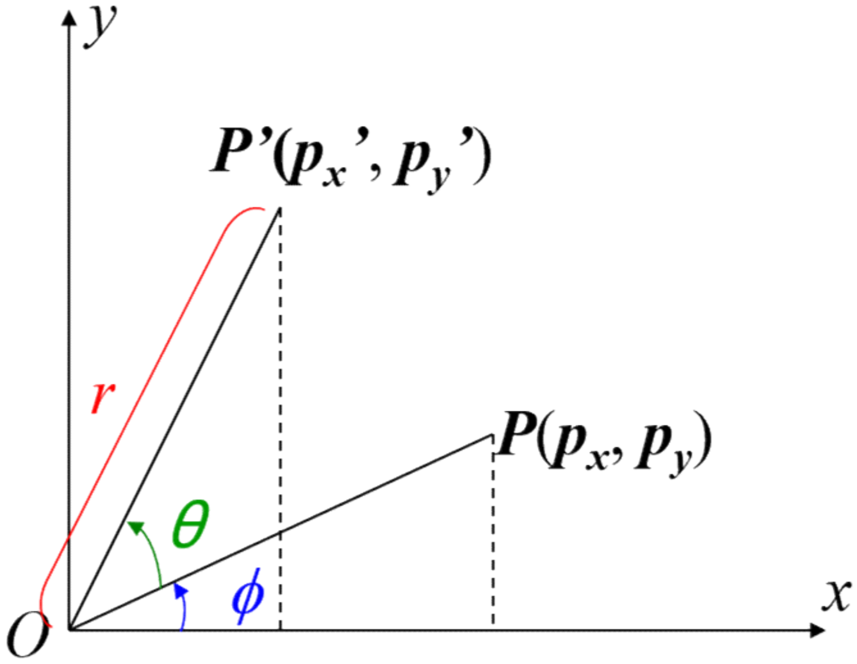
$$P' = P + T = \begin{bmatrix} p_x \\ p_y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

- 旋转变换
  - 绕坐标原点旋转角度$\boldsymbol{\theta}$(逆时针为正，顺时针为负)

$$\begin{cases} p_x' = r\cos(\theta+\phi) \\ p_y' = r\sin(\theta+\phi) \end{cases}$$

$$\begin{cases} p_x' = r\cos\theta\cos\phi - r\sin\theta\sin\phi \\ p_y' = r\cos\theta\sin\phi + r\sin\theta\cos\phi \end{cases}$$

$$\begin{cases} p_x' = p_x\cos\theta - p_y\sin\theta \\ p_y' = p_y\cos\theta + p_x\sin\theta \end{cases}$$



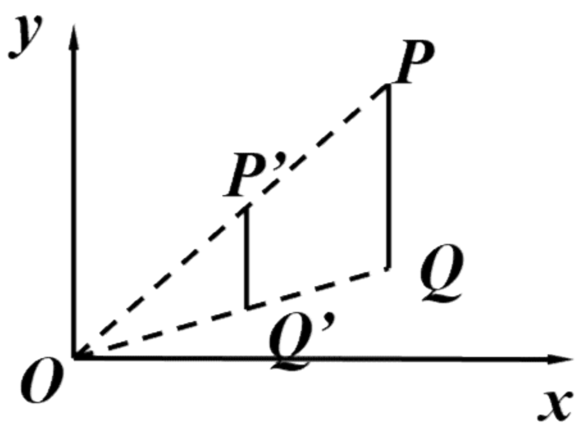
$$P' = R\cdot P = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}\begin{bmatrix} p_x \\ p_y \end{bmatrix}$$

比例(缩放)变换

- 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

$$\begin{cases} p_x' = s_x \cdot p_x \\ p_y' = s_y \cdot p_y \end{cases}$$

$$P' = S \cdot P = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix}$$



- 平移变换 $P' = P + T$ $\quad T = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$

- 旋转变换 $P' = R \cdot P$ $\quad R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$

- 比例变换 $P' = S \cdot P$ $\quad S = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}$

- 为什么需要齐次坐标？

多个变换作用于目标

↓

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

- 齐次坐标
  - $(x, y)^T$点对应的齐次坐标为$(x_h, y_h, h)^T$
    - 其中: $x_h = h \cdot x$, $y_h = h \cdot y$, h为不为0的实数,
    - 通常取h=1,即$(x, y, 1)^T$
    - 如果h=0, $x_h \neq 0$, $y_h \neq 0$, 则$(x_h, y_h, 0)^T$表示无穷远处的点

二维图形的几何变换

- 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

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

- 比例变换

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

- 变换具有统一的表示形式便于变换合成
  - 通过计算变换矩阵的乘积，将任意的变换序列组合成复合变换矩阵

- 复合变换

$$T(t_{x1},t_{y1})\cdot T(t_{x2},t_{y2})=\begin{bmatrix}1&0&t_{x1}\\0&1&t_{y1}\\0&0&1\end{bmatrix}\begin{bmatrix}1&0&t_{x2}\\0&1&t_{y2}\\0&0&1\end{bmatrix}=\begin{bmatrix}1&0&t_{x1}+t_{x2}\\0&1&t_{y1}+t_{y2}\\0&0&1\end{bmatrix}$$

$$=T(t_{x1}+t_{x2},t_{y1}+t_{y2})$$

$$R(\theta_1)\cdot R(\theta_2)=\begin{bmatrix}\cos\theta_1&-\sin\theta_1&0\\\sin\theta_1&\cos\theta_1&0\\0&0&1\end{bmatrix}\begin{bmatrix}\cos\theta_2&-\sin\theta_2&0\\\sin\theta_2&\cos\theta_2&0\\0&0&1\end{bmatrix}$$

$$=\begin{bmatrix}\cos(\theta_1+\theta_2)&-\sin(\theta_1+\theta_2)&0\\\sin(\theta_1+\theta_2)&\cos(\theta_1+\theta_2)&0\\0&0&1\end{bmatrix}=R(\theta_1+\theta_2)$$

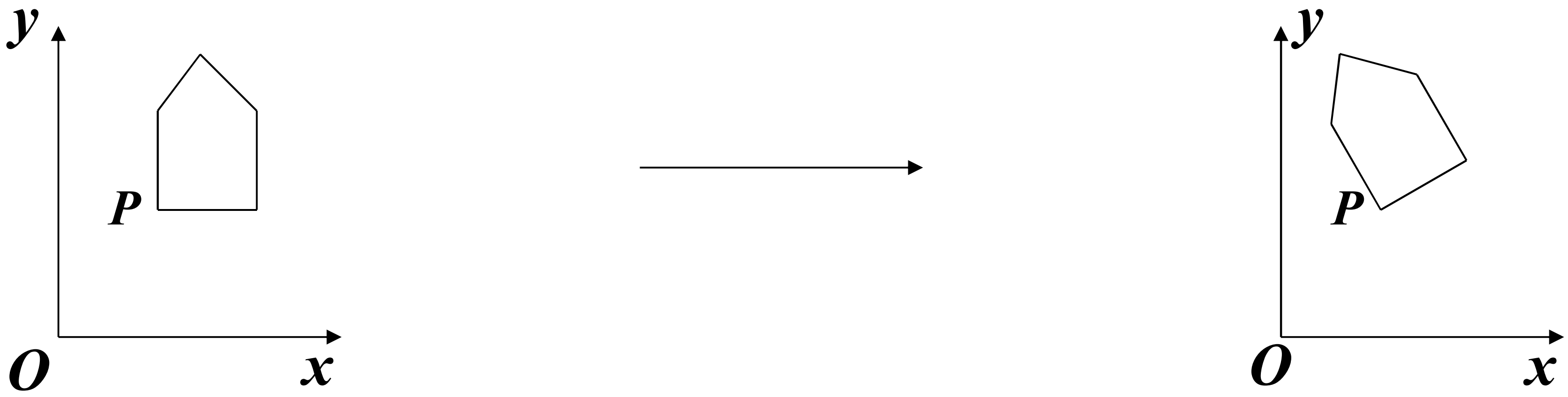
$$S(s_{x1}, s_{y1}) \cdot S(s_{x2}, s_{y2}) = \begin{bmatrix} s_{x1} & 0 & 0 \\ 0 & s_{y1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_{x2} & 0 & 0 \\ 0 & s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_{x1}s_{x2} & 0 & 0 \\ 0 & s_{y1}s_{y2} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= S(s_{x1}s_{x2}, s_{y1}s_{y2})$$

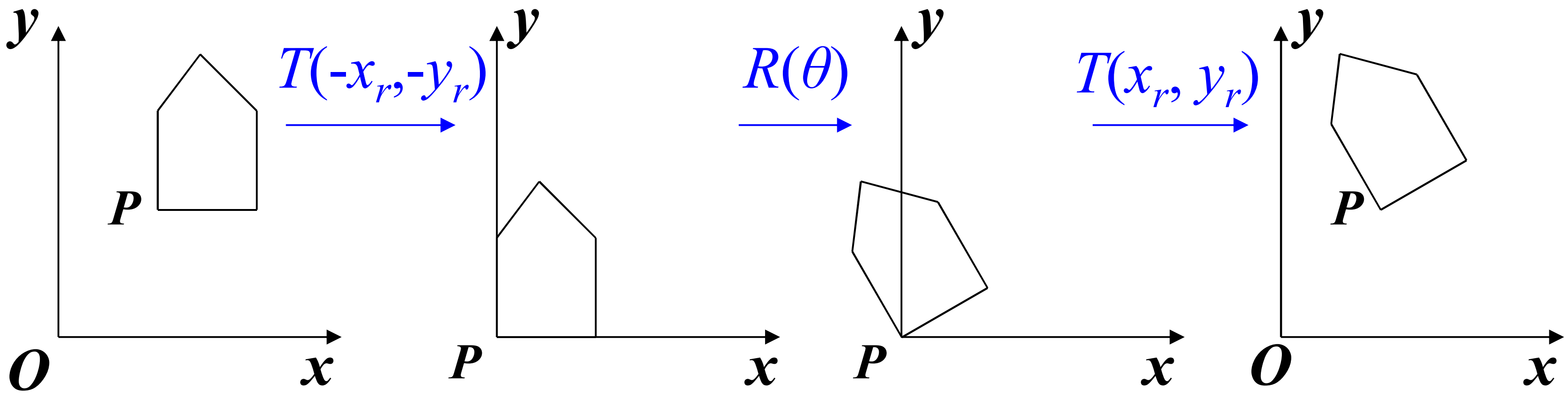
- 问题：如何实现复杂变换？

变换分解 → 变换合成

- 关于任意点$P(x_r, y_r)$的旋转变换



- 关于任意点$P(x_r, y_r)$的旋转变换

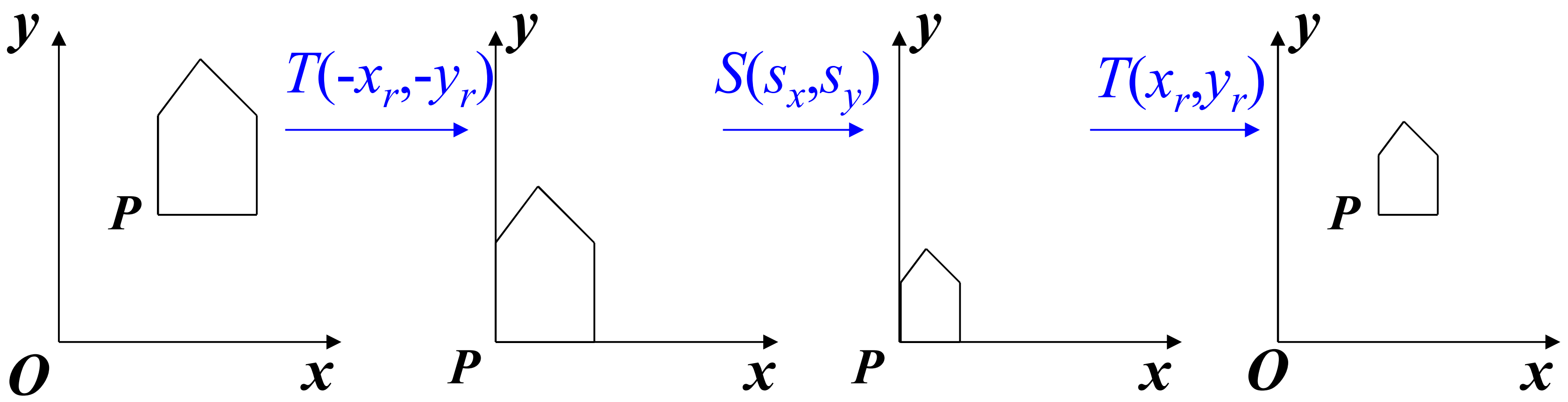


$$T = T(x_r, y_r)\cdot R(\theta)\cdot T(-x_r,-y_r)$$

$$=\begin{bmatrix}1 & 0 & x_r\\ 0 & 1 & y_r\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & -x_r\\ 0 & 1 & -y_r\\ 0 & 0 & 1\end{bmatrix}$$

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

- 关于任意点$P(x_r, y_r)$的比例变换



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

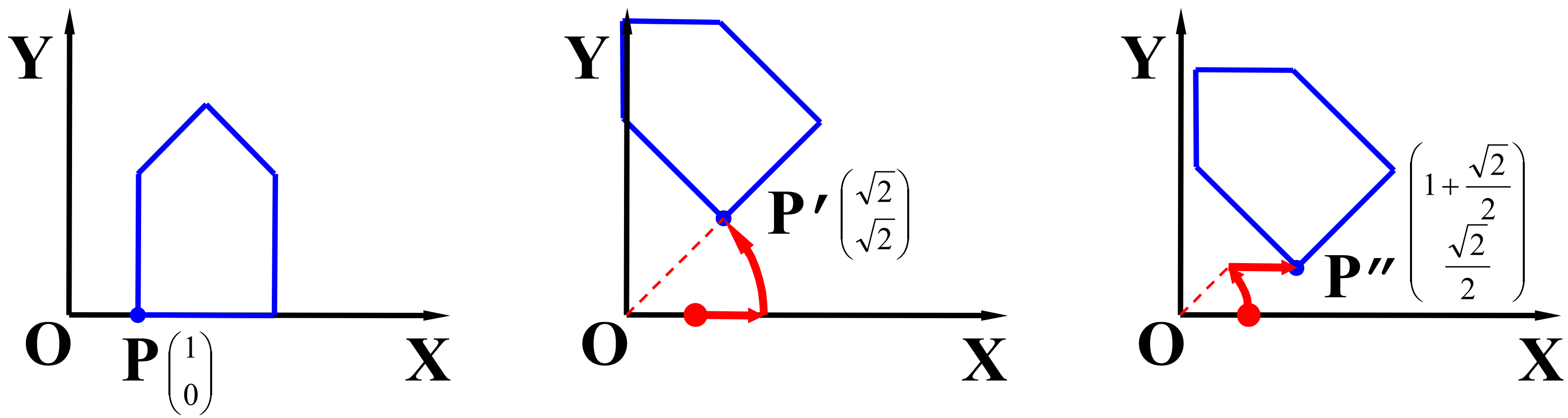
$$= \begin{bmatrix} 1 & 0 & x_r \\ 0 & 1 & y_r \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & -x_r \\ 0 & 1 & -y_r \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} s_x & 0 & (1-s_x)x_r \\ 0 & s_y & (1-s_y)y_r \\ 0 & 0 & 1 \end{bmatrix}$$

- 复合变换：结果与变换的顺序有关

先平移变换
后旋转变换

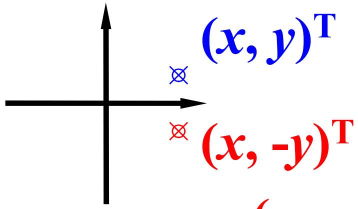
后旋转变换
先平移变换



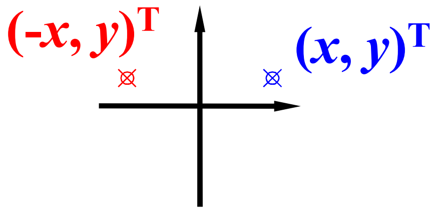
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & 0 \\ d & e & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

对称(反射)变换

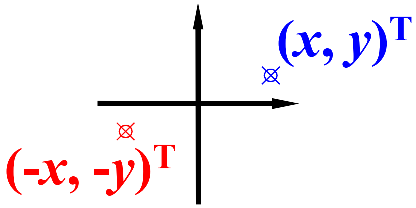
- 关于x轴：$b=d=0,\ a=1,\ e=-1$



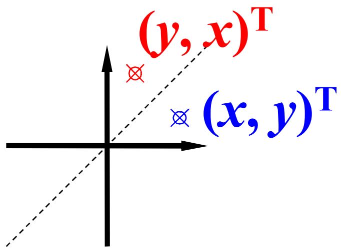
- 关于y轴：$b=d=0,\ a=-1,\ e=1$



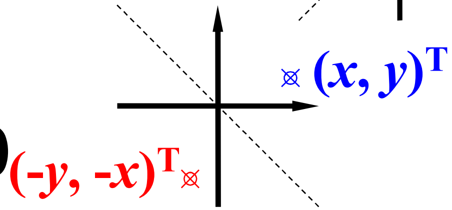
- 关于原点：$b=d=0,\ a=e=-1$



- 关于直线$y=x$：$b=d=1,\ a=e=0$



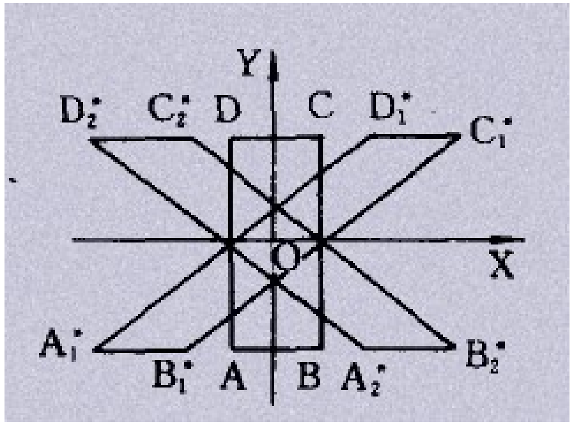
- 关于直线$y=-x$：$b=d=-1,\ a=e=0$

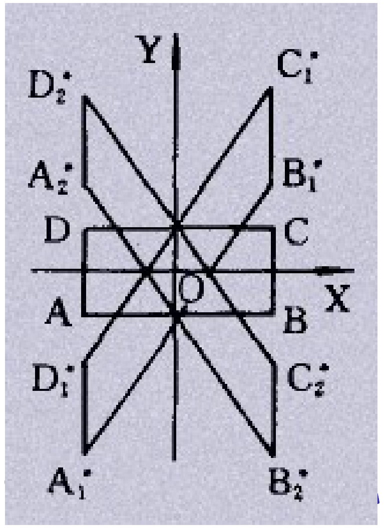


- 错切变换

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & b & 0 \\ d & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

  - 沿$\boldsymbol{x}$方向错切： $\boldsymbol{d=0}$
  - 沿$\boldsymbol{y}$方向错切： $\boldsymbol{b=0}$
  - 沿$\boldsymbol{x, y}$方向错切： $\boldsymbol{b\neq 0}$且$\boldsymbol{d\neq 0}$





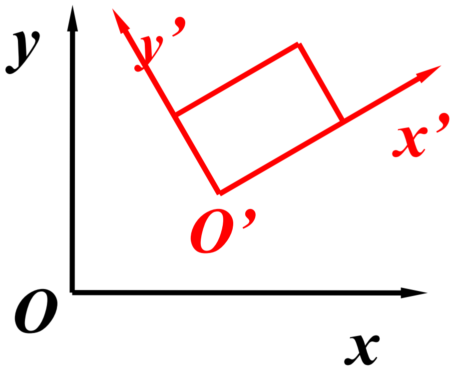
二维变换

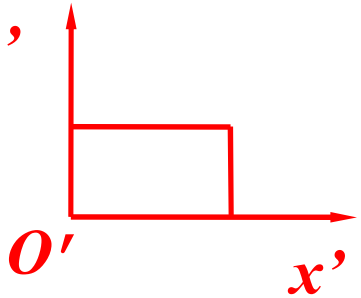
- 仿射变换

$$\begin{cases} x' = ax + by + e \\ y' = cx + dy + f \end{cases} \quad \Longleftrightarrow \quad A_f = \begin{bmatrix} a & b & e \\ c & d & f \\ 0 & 0 & 1 \end{bmatrix}$$

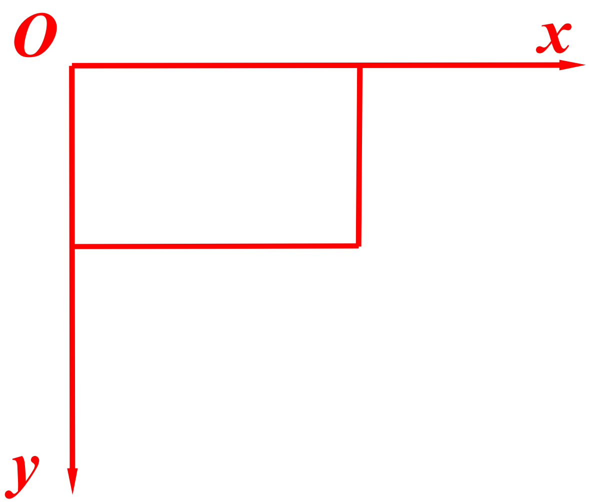
二维图形的显示流程

- 坐标系(**Coordinate System**)：建立了图形与数之间的对应联系
  - 世界坐标系(**WCS**) /用户坐标系(**UCS**)
  - 局部坐标系(**LCS**) /造型坐标系(**MCS**)



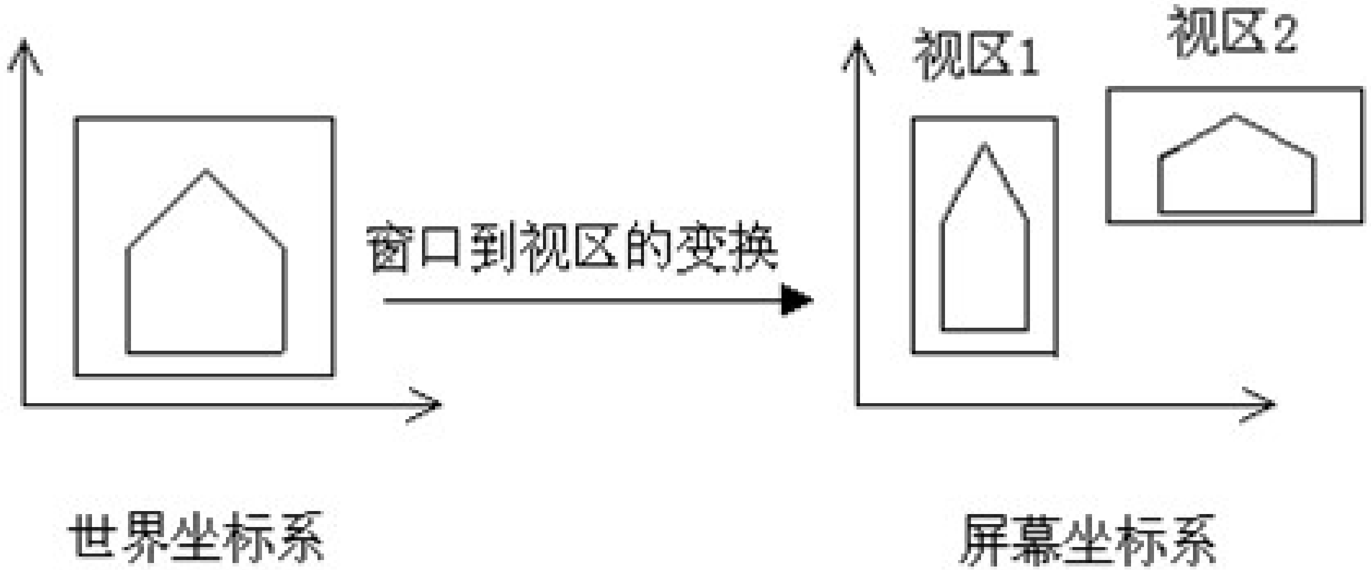


- 屏幕坐标系(Screen Coordinate System) / 设备坐标系(Device Coordinate System)



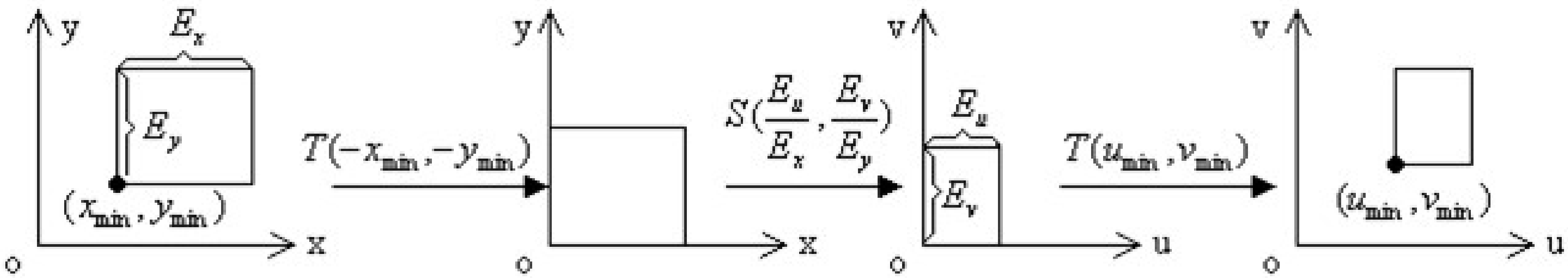
- 窗口
  - 在世界坐标系中指定的矩形区域
  - 用来指定要显示的图形
- 视区(视口)
  - 在设备坐标系（屏幕或绘图纸）上指定的矩形区域
  - 用来指定窗口内的图形在屏幕上显示的大小及位置
- 窗口到视区的变换——观察变换


世界坐标系
关于窗口的裁剪
窗口到视区的变换
设备坐标系
扫描转换
显示

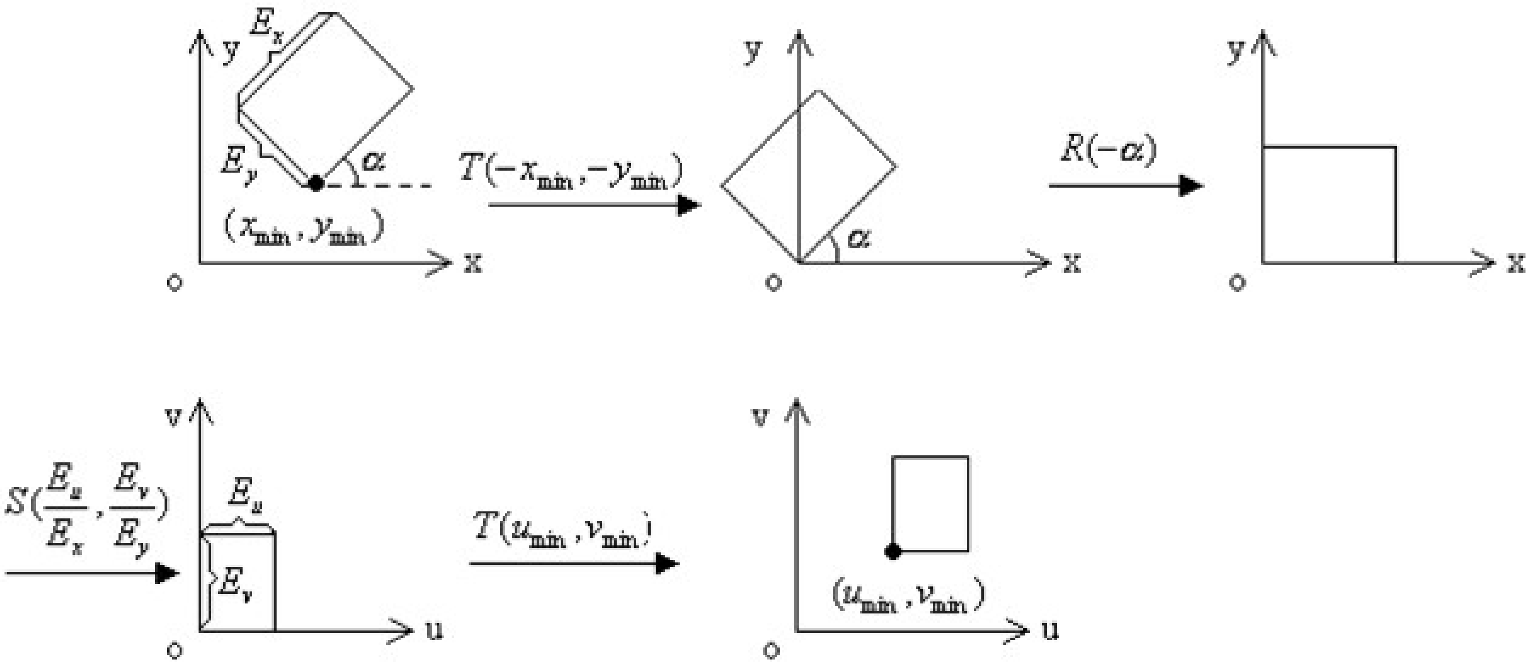

窗口到视区的变换
视区1
视区2
世界坐标系
屏幕坐标系

窗口到视区的变换

- 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

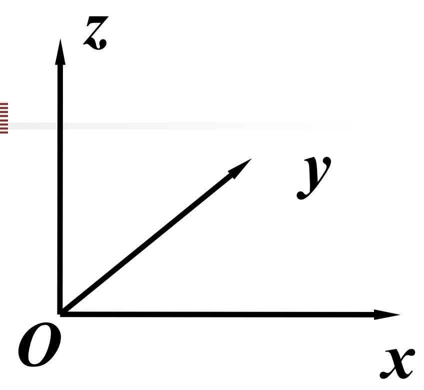


$$M_{wv} = T(u_{min}, v_{min})S\left(\frac{E_u}{E_x}, \frac{E_v}{E_y}\right)T(-x_{min}, -y_{min})$$



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

三维几何变换



- 三维齐次坐标
  - $(x, y, z)^T$点对应的齐次坐标为$(x_h, y_h, z_h, h)^T$
    - 其中: $x_h = h \cdot x$, $y_h = h \cdot y$, $z_h = h \cdot z$, h为不为0的实数
    - 通常取$h=1$,即$(x, y, z, 1)^T$
    - 如果$h=0$，则该坐标表示无穷远处的点

- 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

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

  - 以坐标原点为中心参照点

- 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

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

- 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
  - 以z轴为依赖轴

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

- 对称变换
  - 关于坐标平面***xy***的对称变换

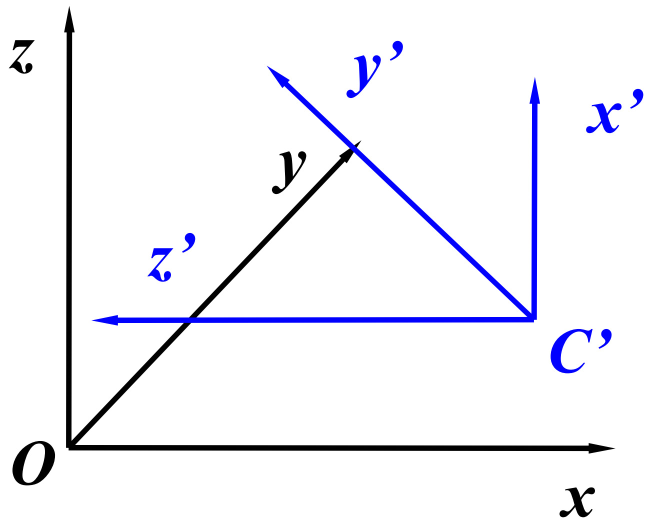
$$SY_{xy}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&-1&0\\0&0&0&1\end{bmatrix}$$

- 三维变换的一般形式

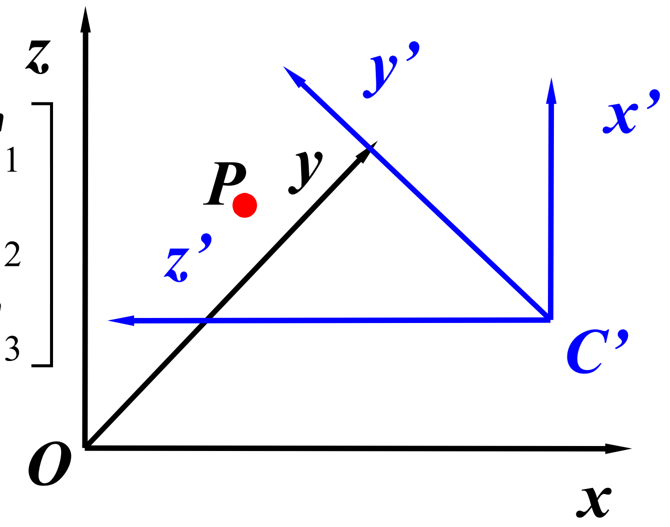
$$A=\begin{bmatrix}a_{11}&a_{12}&a_{13}&a_{14}\\a_{21}&a_{22}&a_{23}&a_{24}\\a_{31}&a_{32}&a_{33}&a_{34}\\0&0&0&1\end{bmatrix}$$

三维坐标系之间的变换

- 求三维点*P*在新坐标系下的坐标值
- 在新坐标系下的坐标值反求原坐标值
  - 提示： 参考二维情形



- 在坐标系***Oxyz***下，点***P***的坐标为 $P=\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$
  - 点 $C'=\begin{bmatrix} c'_x \\ c'_y \\ c'_z \end{bmatrix}$
  - 单位向量 $X'=\begin{bmatrix} x'_1 \\ x'_2 \\ x'_3 \end{bmatrix}$ $Y'=\begin{bmatrix} y'_1 \\ y'_2 \\ y'_3 \end{bmatrix}$ $Z'=\begin{bmatrix} z'_1 \\ z'_2 \\ z'_3 \end{bmatrix}$

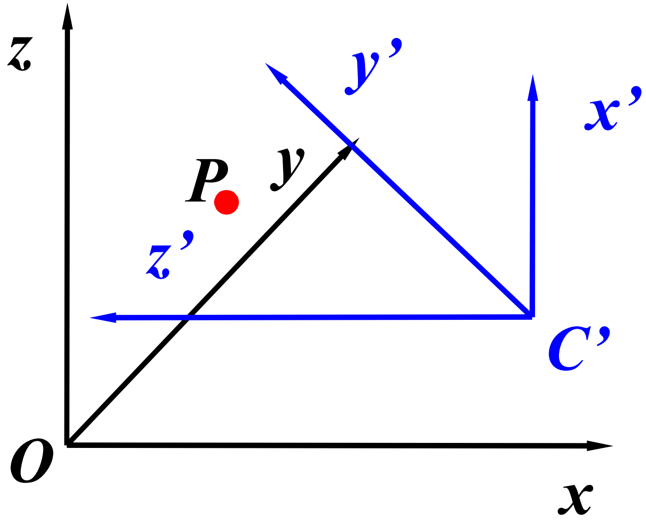


- 在坐标系***C'x'y'z'***下：
  - ➢点***P***的新坐标***P'*** $P'=\begin{bmatrix} p_x' \\ p_y' \\ p_z' \end{bmatrix}$

➢从***P*** ➔***P'***?

➢从***P'***➔ ***P*** ?

- 从$P \rightarrow P'$



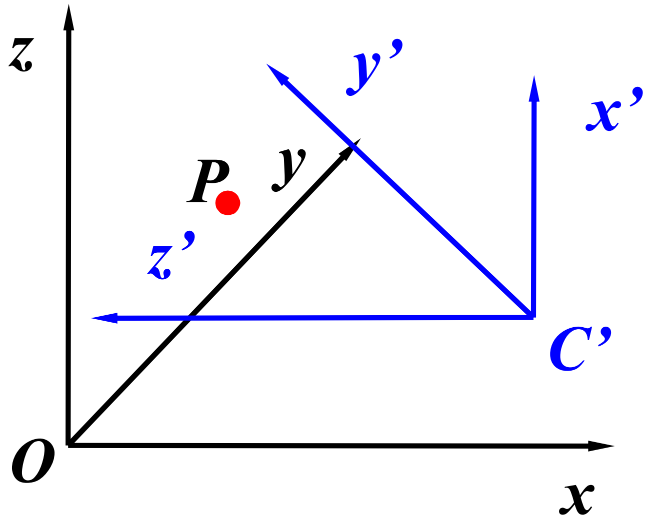
$$P' = [p_x' \quad p_y' \quad p_z' \quad 1]^T = R \cdot T \cdot P$$

$$= \begin{bmatrix} x_1' & x_2' & x_3' & 0 \\ y_1' & y_2' & y_3' & 0 \\ z_1' & z_2' & z_3' & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -c_x' \\ 0 & 1 & 0 & -c_y' \\ 0 & 0 & 1 & -c_z' \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix}$$

或

$$P' = \begin{bmatrix} p_x' \\ p_y' \\ p_z' \end{bmatrix} = \begin{bmatrix} (P - C') \cdot X' \\ (P - C') \cdot Y' \\ (P - C') \cdot Z' \end{bmatrix}$$

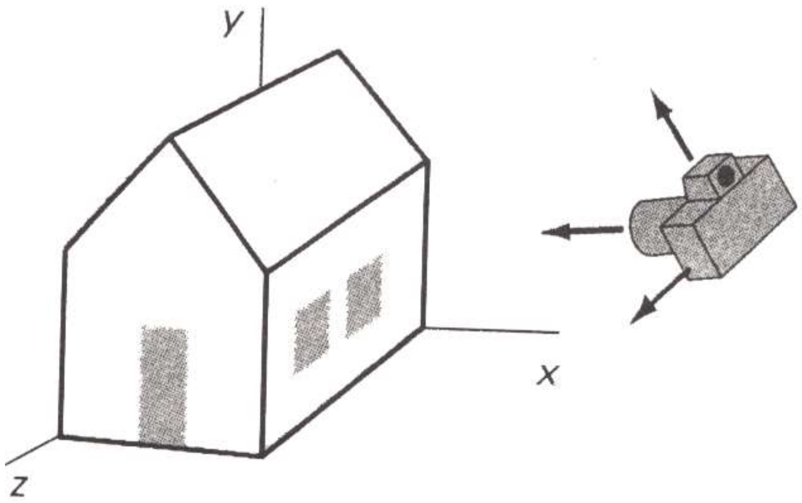
- 从P' ➔ P



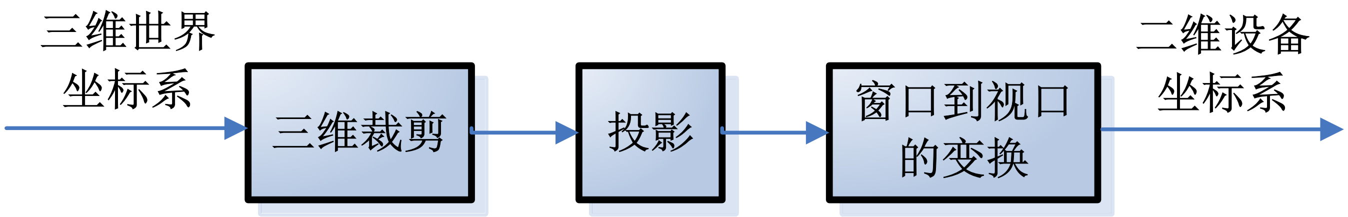
$$P=\begin{bmatrix}p_x\\p_y\\p_z\end{bmatrix}=C'+\left(p_x'X'+p_y'Y'+p_z'Z'\right)$$

投影变换

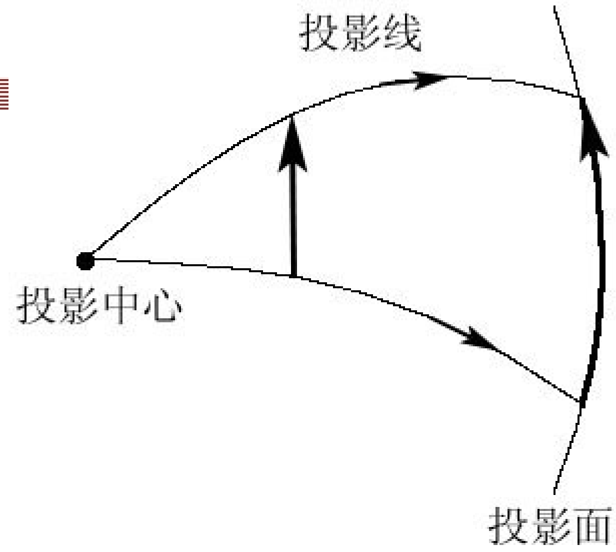
- 投影变换：三维物体转换为二维图形表示的过程

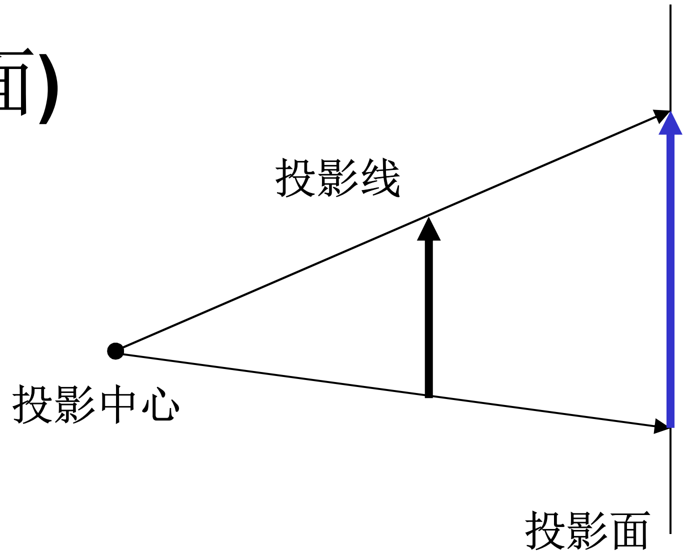


- 投影——照相机模型
  - 选定投影类型
  - 设置投影参数(观察点、观察方向、观察平面/投影平面) ↔ 相机位置、方向、胶片所在平面等
  - 三维裁剪 ↔ 取景
  - 投影和显示 ↔ 成像
- 简单的三维图形显示过程：



- 平面几何投影及其分类
  - 投影
    - 将**n**维的点变换成小于**n**维的点
    - 将**3**维的点变换成**2**维的点
  - 投影中心(观察点/视点)
  - 投影平面(投影面/观察平面)
  - 投影线
  - 平面几何投影
    - 投影面是平面
    - 投影线为直线

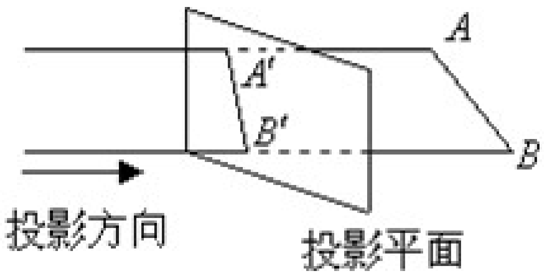


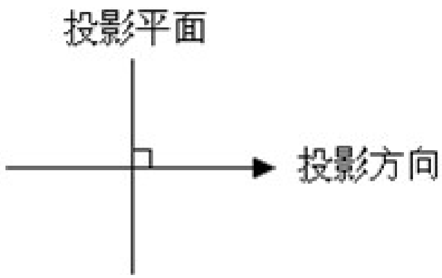


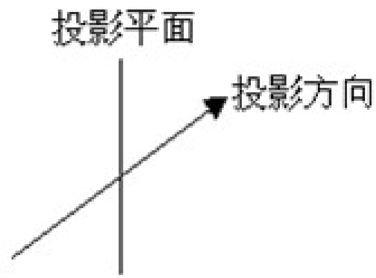
投影变换分类

- 投影变换
  - 平行投影
    - 正平行投影
      - 正投影
      - 正轴测投影
        - 正等测
        - 正二测
        - 正三测
    - 斜平行投影
      - 斜等测
      - 斜二测
  - 透视投影
    - 一点透视
    - 二点透视
    - 三点透视

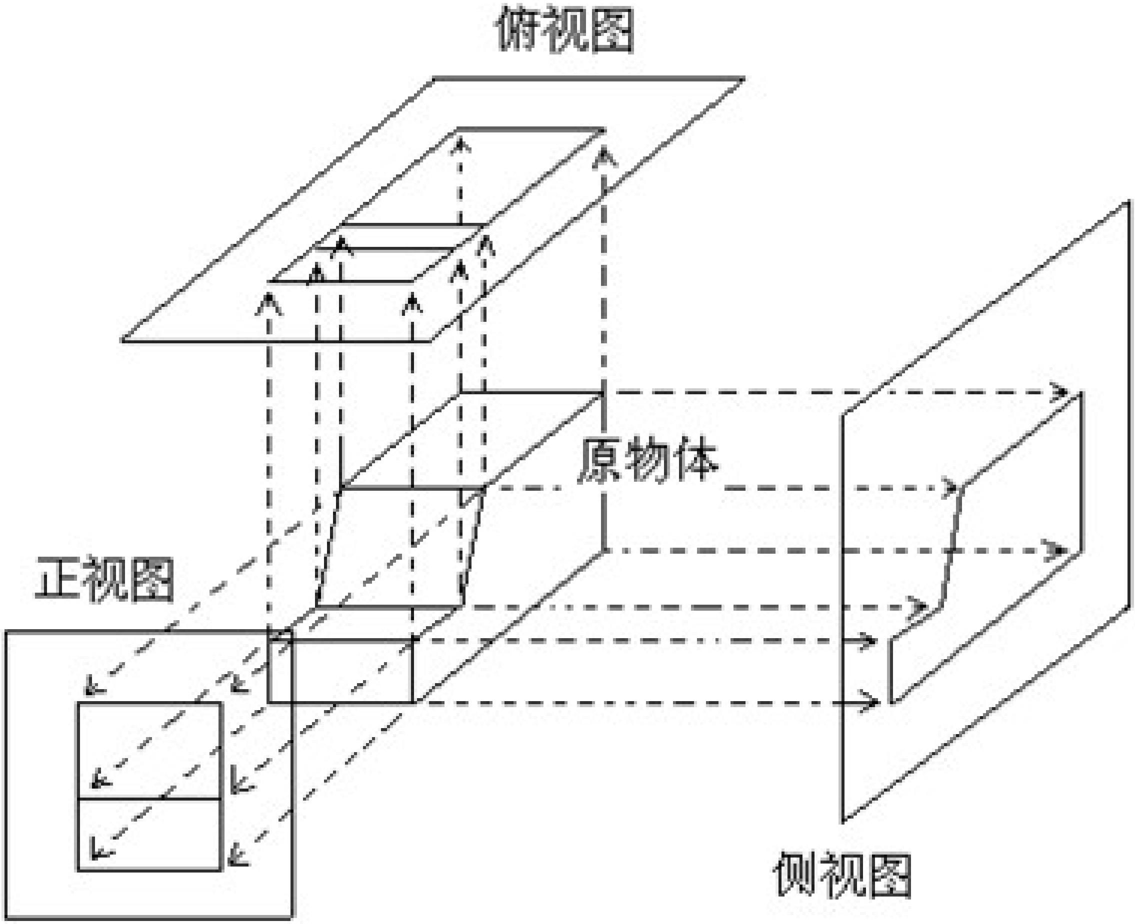
- 平行投影：投影中心与投影平面间距离无穷远
  - 投影方向与投影平面法向间的关系
    - 正平行投影
    - 斜平行投影



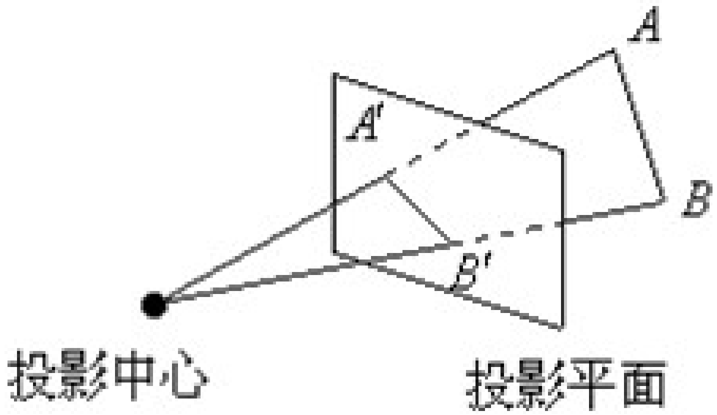


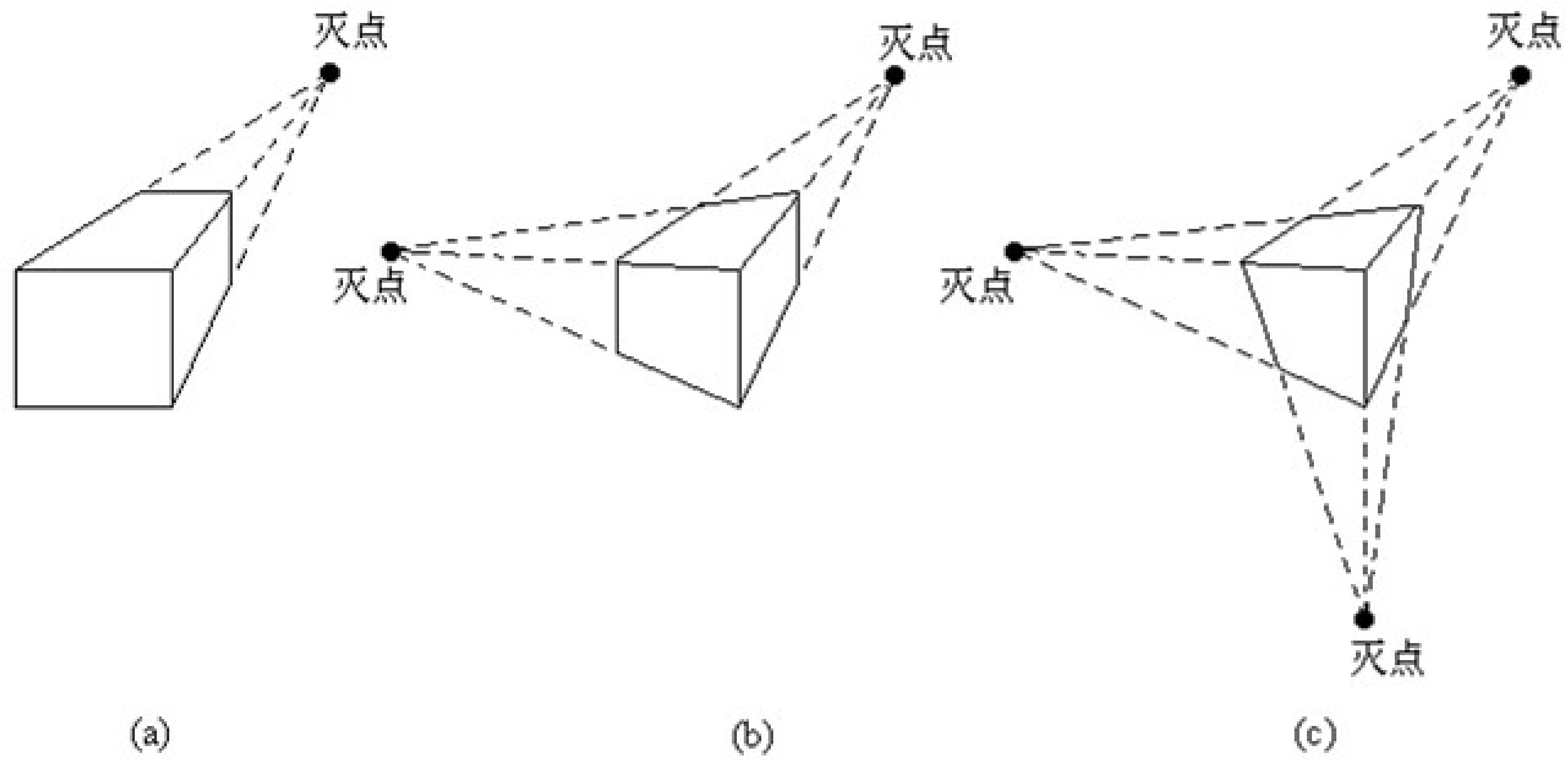


■ 三视图：正视图、侧视图和俯视图



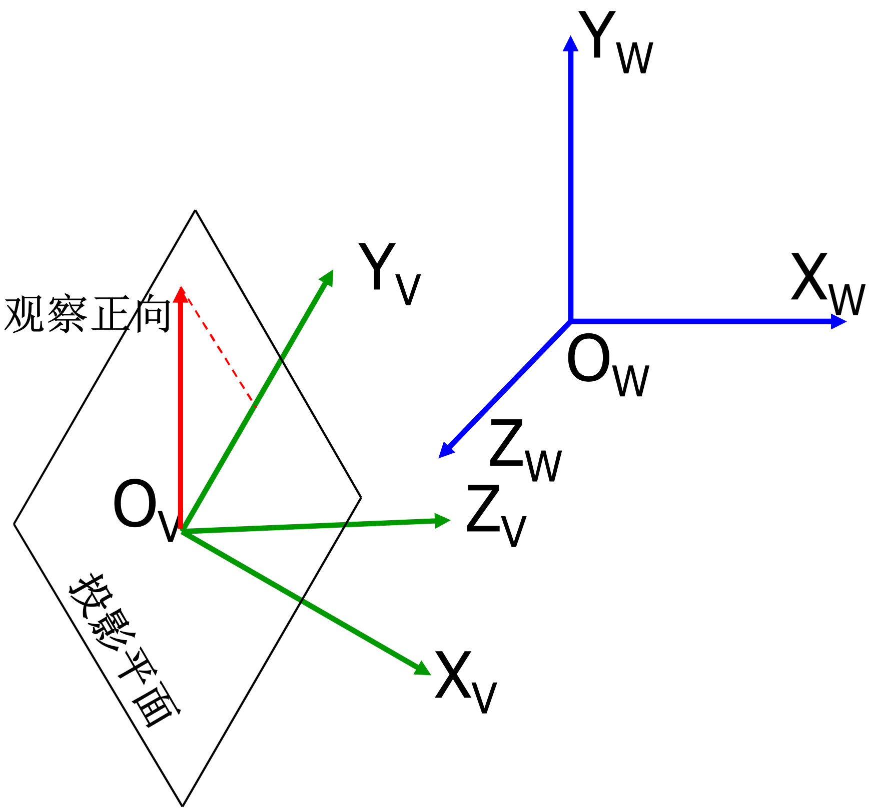
- 透视投影：投影中心与投影平面间距离有限
  - 灭点：不平行于投影平面的平行线，经过透视投影之后收敛于一点
  - 主灭点：平行于坐标轴的平行线的灭点

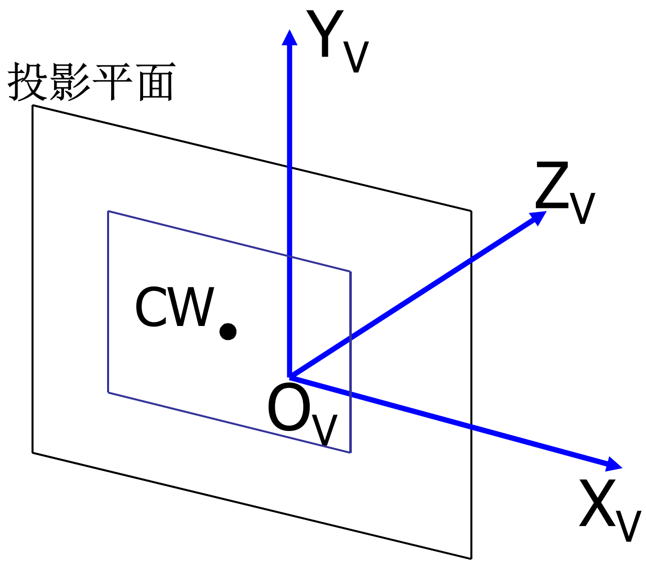




灭点
灭点
灭点
灭点
灭点
灭点
(a)
(b)
(c)

观察坐标系



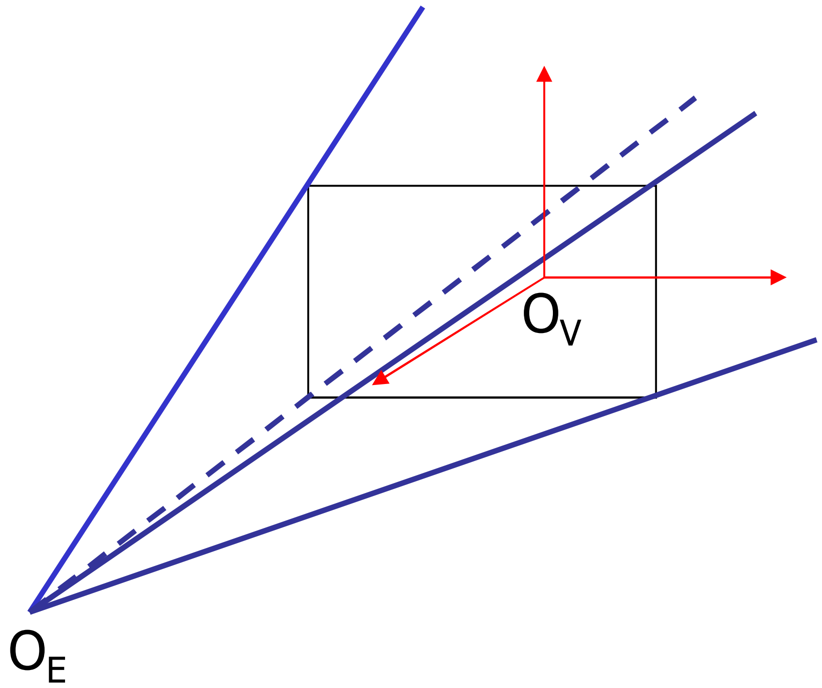


窗口左下角：(x_{vmin}, y_{vmin})
窗口右上角：(x_{vmax}, y_{vmax})

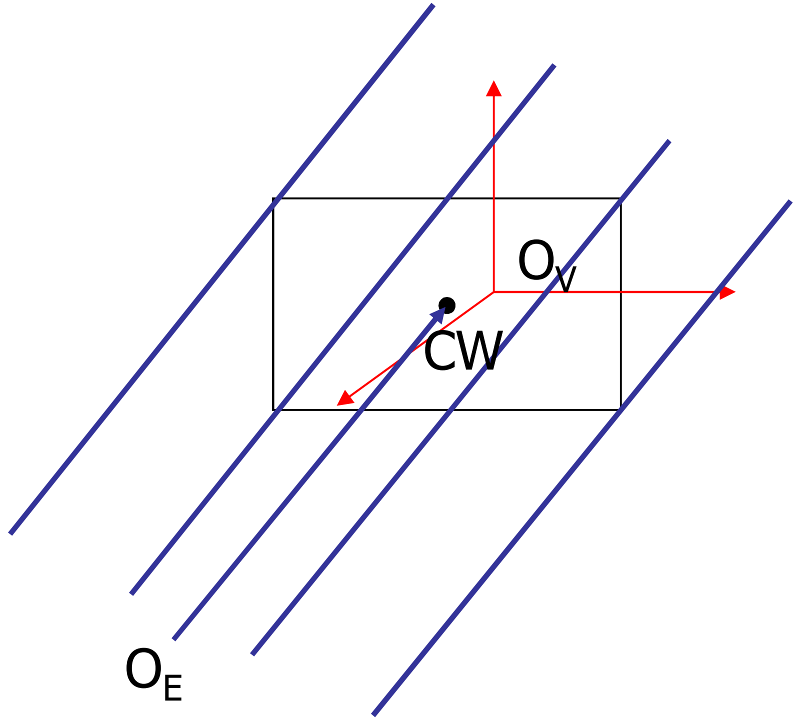
- 从世界坐标系到观察坐标系的变换
  - 观察坐标系的原点在世界坐标系下的位置$(\boldsymbol{O_{vx}}, \boldsymbol{O_{vy}}, \boldsymbol{O_{vz}})$
  - 观察坐标系的三个坐标轴向量分别为：$(\boldsymbol{X_{vx}}, \boldsymbol{X_{vy}}, \boldsymbol{X_{vz}}), (\boldsymbol{Y_{vx}}, \boldsymbol{Y_{vy}}, \boldsymbol{Y_{vz}}), (\boldsymbol{Z_{vx}}, \boldsymbol{Z_{vy}}, \boldsymbol{Z_{vz}})$
  - 变换矩阵$\boldsymbol{M_{W\to V}=R\cdot T}$

$$T=\begin{bmatrix}1 & 0 & 0 & -O_{vx}\\ 0 & 1 & 0 & -O_{vy}\\ 0 & 0 & 1 & -O_{vz}\\ 0 & 0 & 0 & 1\end{bmatrix}\quad R=\begin{bmatrix}X_{vx} & X_{vy} & X_{vz} & 0\\ Y_{vx} & Y_{vy} & Y_{vz} & 0\\ Z_{vx} & Z_{vy} & Z_{vz} & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

投影空间

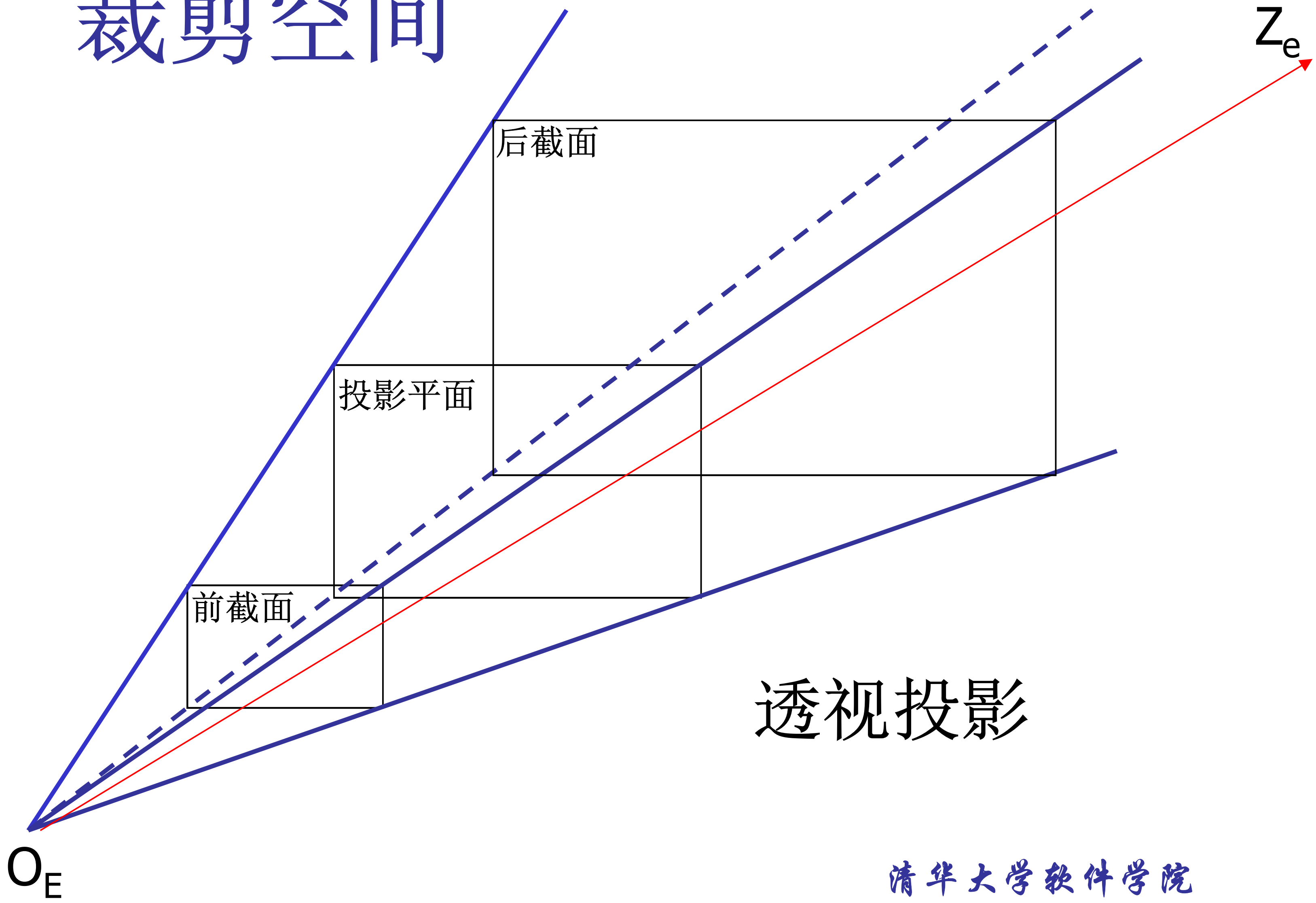


透视投影



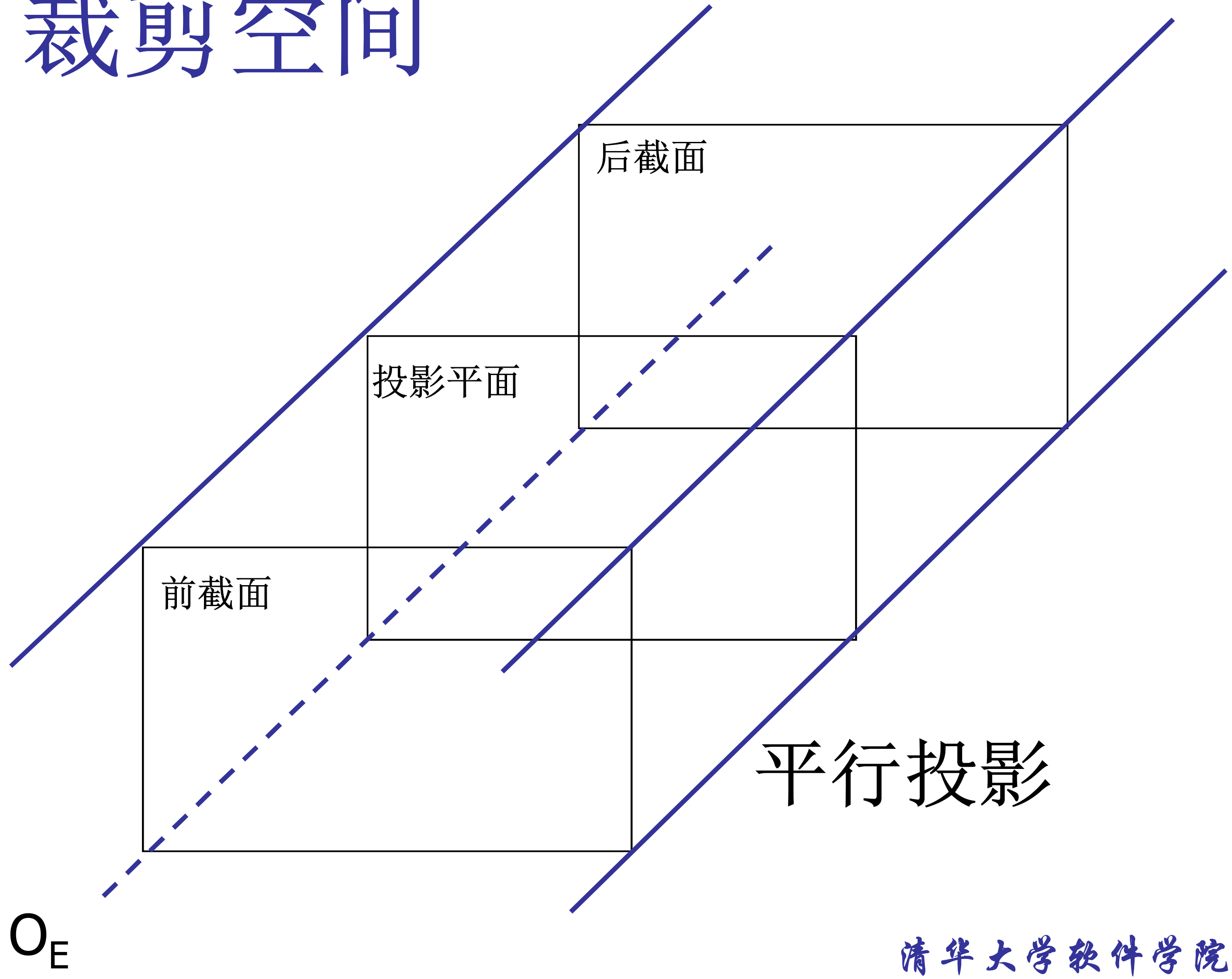
平行投影

裁剪空间



透视投影

裁剪空间



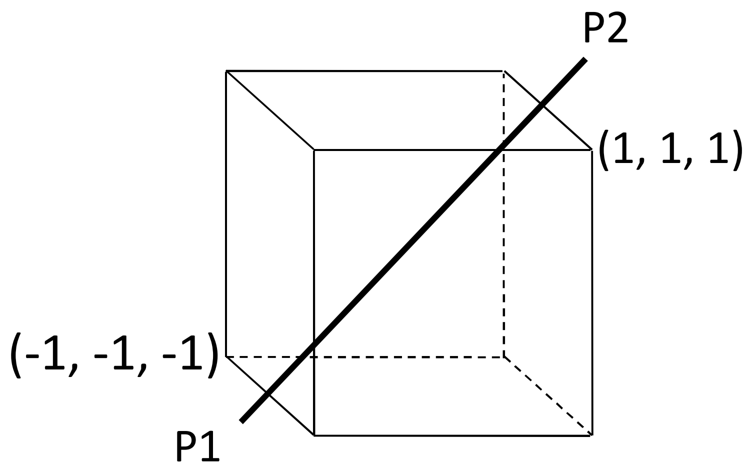
三维裁剪

- 二维裁剪算法的推广
  - 直线段裁剪的**Cohen-Sutherland**算法、梁-**Barskey**算法的直接推广
  - 多边形裁剪的**Sutherland-Hodgman**算法的直接推广
- 裁剪在规范的裁剪空间进行

- 直线段的三维裁剪

$P_1(-2, -1, 1/2)$

$P_2(3/2, 3/2, -1/2)$



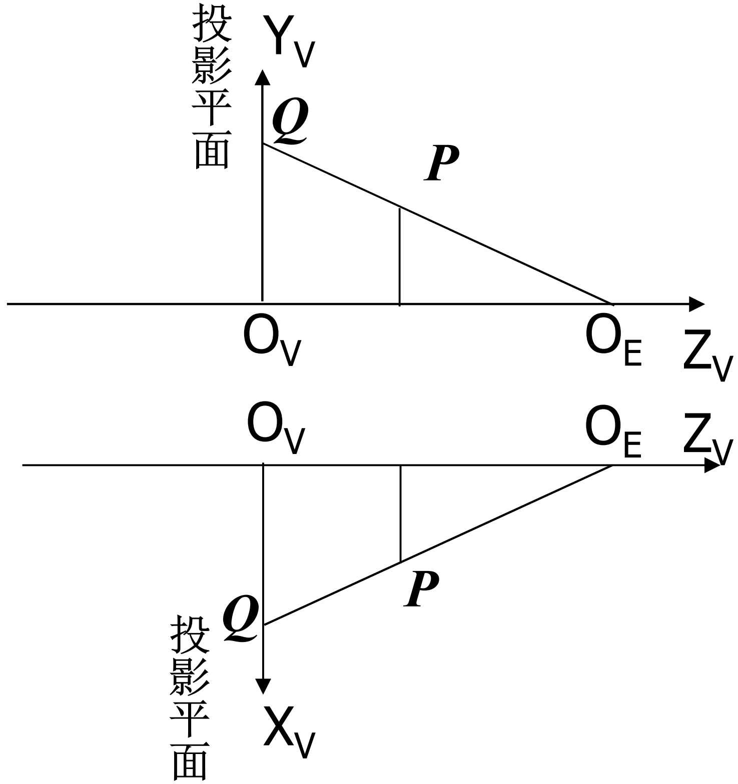
$$x_L \le x_1+(x_2-x_1)t \le x_R$$

$$y_B \le y_1+(y_2-y_1)t \le y_F$$

$$z_D \le z_1+(z_2-z_1)t \le z_T$$

透视投影变换

- 在观察坐标系$O_VX_VY_VZ_V$中，投影平面为$Z_V=0$，投影中心为$(0, 0, d)$，待投影点为$P(x_P, y_P, z_P)$，求投影点$Q(x_Q, y_Q, z_Q)$



$$
\begin{cases}
x_Q = \dfrac{x_P}{1-(z_P/d)} \\
y_Q = \dfrac{y_P}{1-(z_P/d)} \\
z_Q = 0
\end{cases}
$$

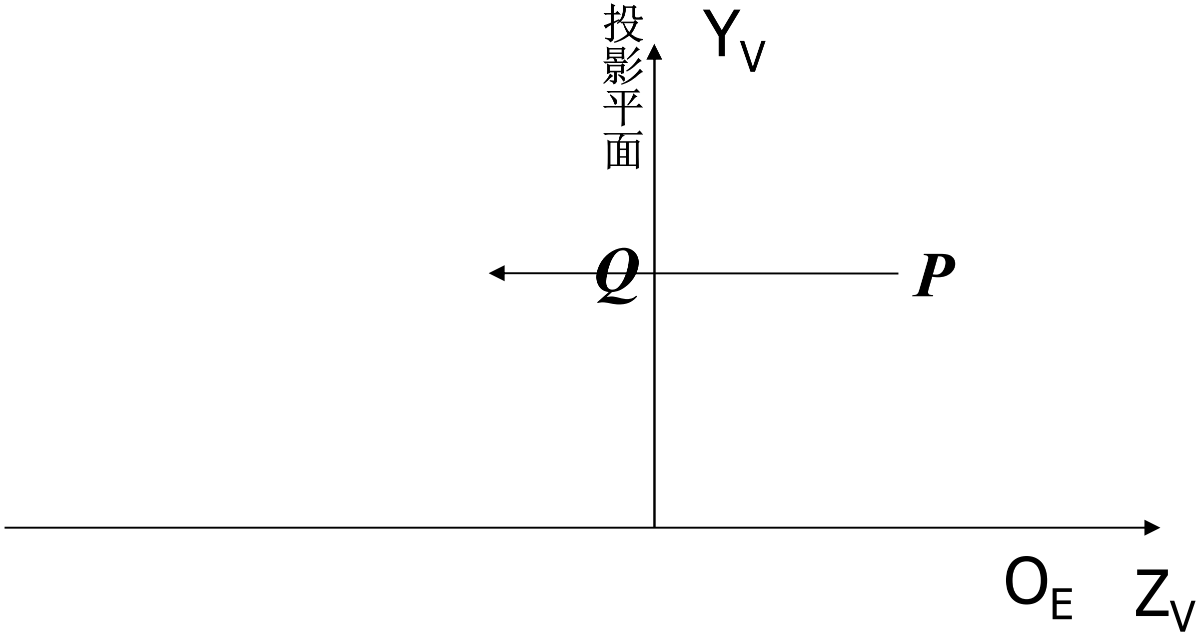
- 透视投影变换矩阵

$$M_{per} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\dfrac{1}{d} & 1 \end{bmatrix}$$

$$Q = M_{per} \cdot P$$

平行投影变换

- 在观察坐标系$O_VX_VY_VZ_V$中，投影平面为Z_V=0，投影方向为(0, 0, -1)，待投影点为$P(x_P, y_P, z_P)$，求投影点$Q(x_Q, y_Q, z_Q)$



$$
\begin{cases}
x_Q = x_P \\
y_Q = y_P \\
z_Q = 0
\end{cases}
$$

- 平行投影变换矩阵

$$M_{ort}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&1\end{bmatrix}\qquad M_{per}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&-\frac{1}{d}&1\end{bmatrix}$$

$$Q=M_{ort}\cdot P$$

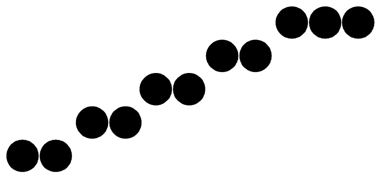
- 透视投影与平行投影之间的关系

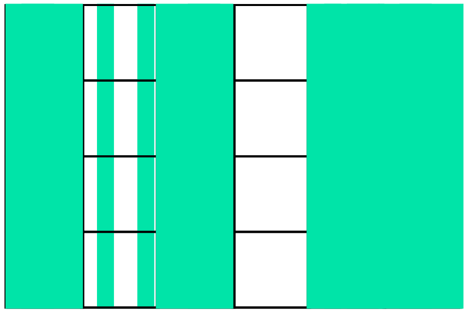
$$\lim_{d\to\infty}M_{per}=M_{ort}$$

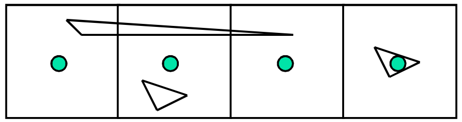
三维图形的显示流程

反走样

- 用离散量表示连续量引起的失真现象称之为走样(**aliasing**)



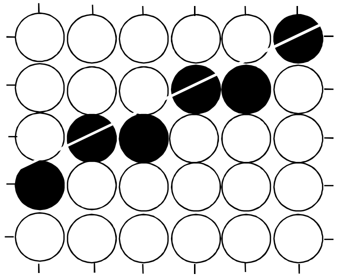


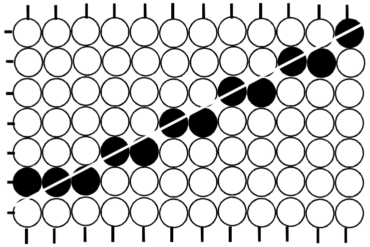


- 用于减少或消除失真效果的技术称为反走样(**antialiasing**)

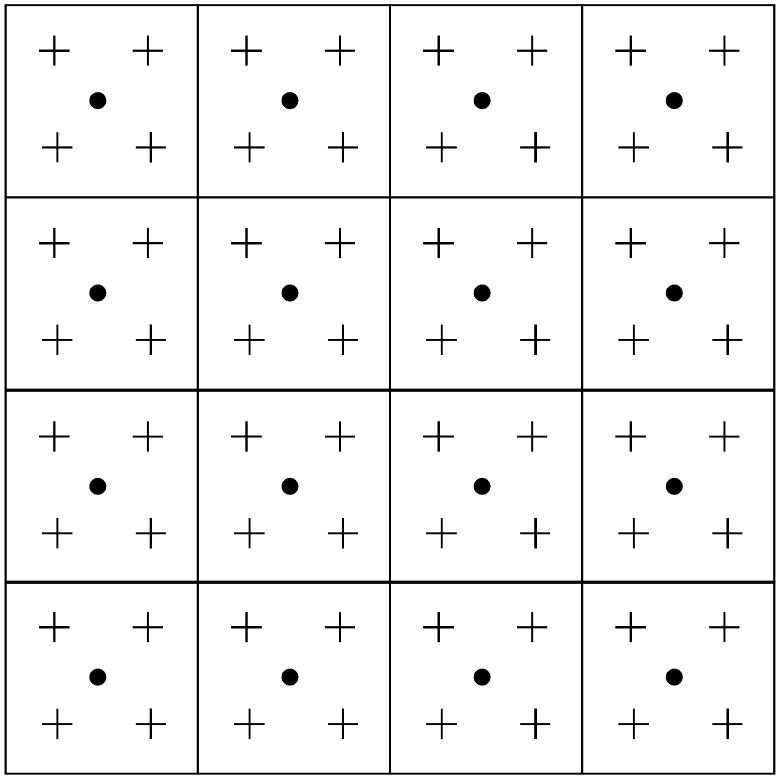
提高分辨率

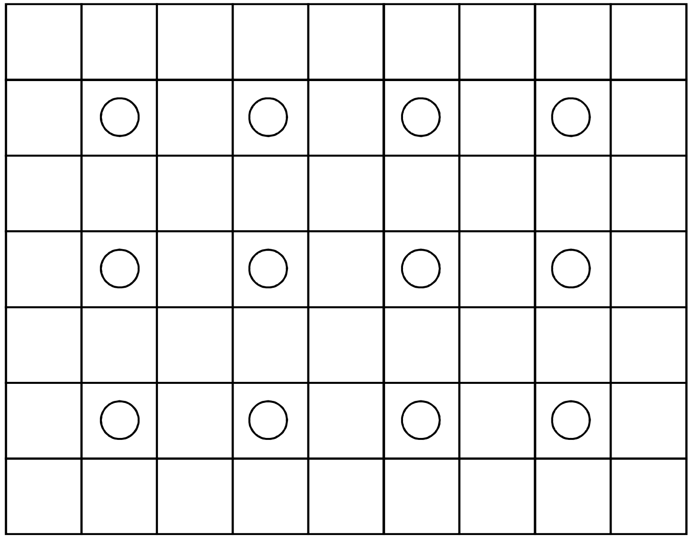
- 提高显示器分辨率
  - 直线经过两倍的象素，锯齿也增加一倍；
  - 每个锯齿的宽度也减小了一倍；
  - 显示出的直线段看起来就平直光滑了一些。
- 增加分辨率虽然简单，但不经济
- 只能减轻而不能消除锯齿问题





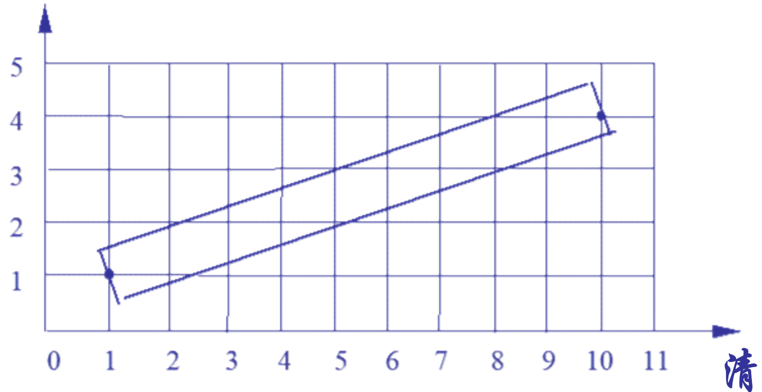
超采样



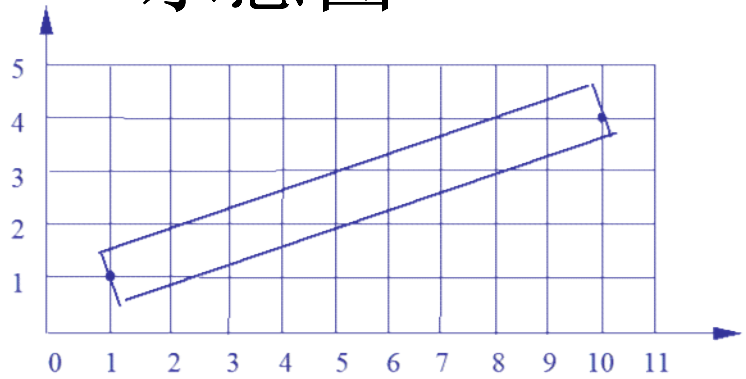


区域采样

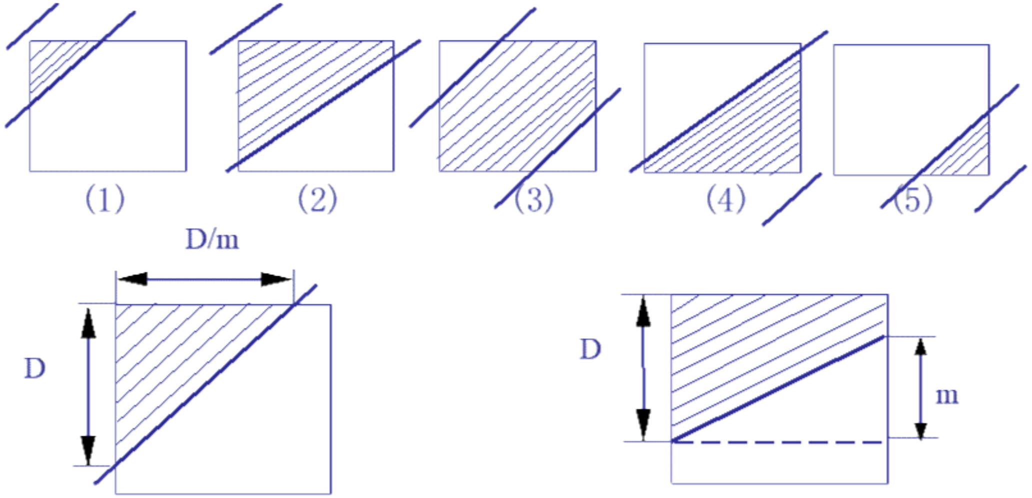
- 基本思想：
  - 每个象素是一个具有一定面积的小区域，将直线段看作具有一定宽度的狭长矩形。当直线段与象素有交时，求出两者相交区域的面积，然后根据相交区域面积的大小确定该象素的亮度值。



示意图



有宽度的线条轮廓

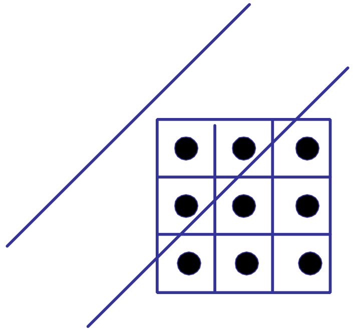


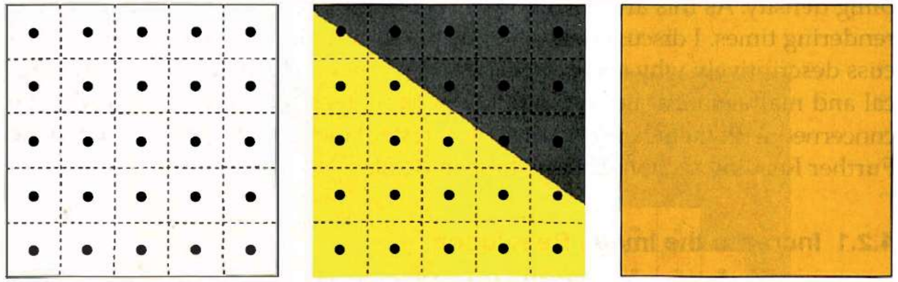
象素相交的五种情况及用于计算面积的量

面积计算

- 情况**(1)(5)**阴影面积为：$D^2/(2m)$；
- 情况**(2)(4)**阴影面积为：$D - m/2$；
- 情况**(3)**阴影面积为：$1 - D^2/m$

- 为了简化计算可以采用离散的方法



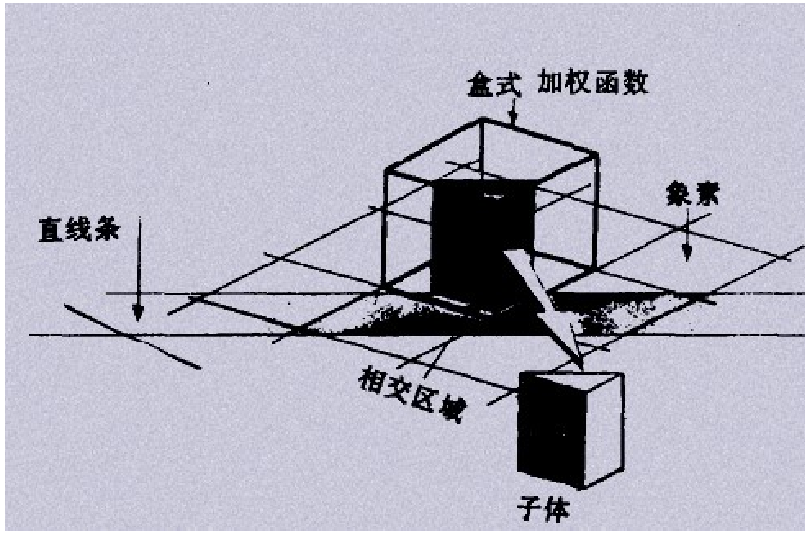


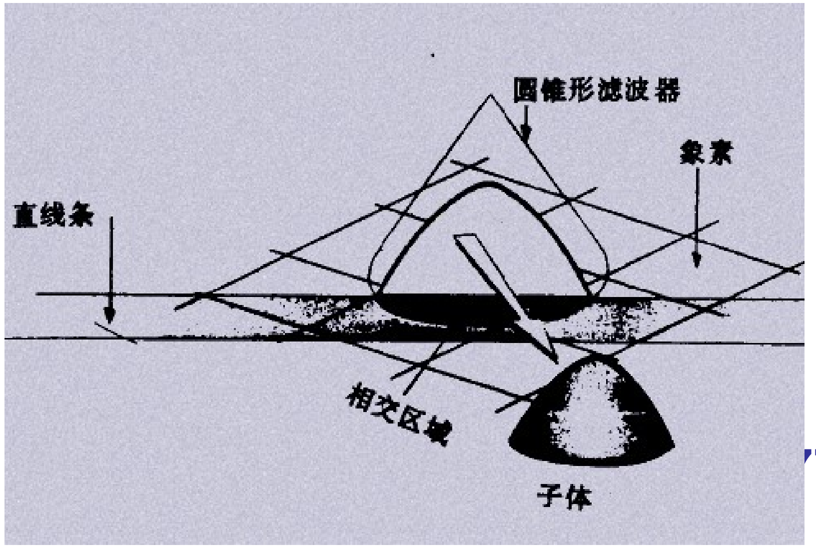
n=9,k=3近似面积为1/3

  - 将屏幕象素均分成n个子象素；
  - 计算中心点落在直线段内的子象素的个数k；
  - 屏幕该象素的亮度取决于相交区域面积的近似值k/n。
- 非加权区域采样方法的缺点

加权区域采样

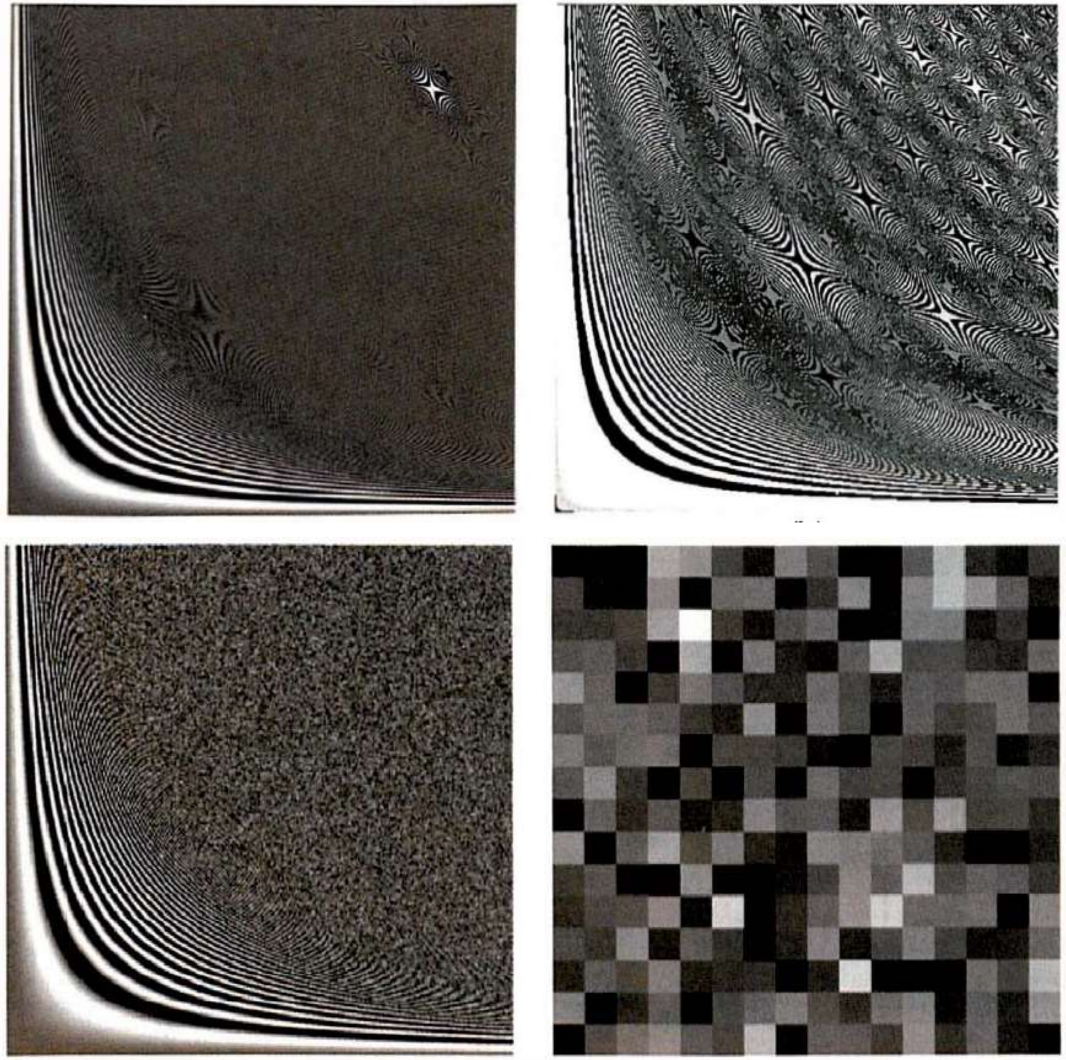
- 基本思想：
  - 相交区域对象素亮度的贡献还依赖于该区域与象素中心的距离；
  - 当直线经过某象素时，该象素的亮度是在两者相交区域上对圆锥形滤波器进行积分的积分值。



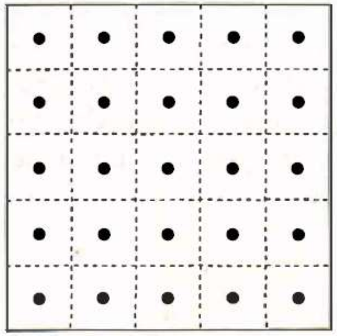


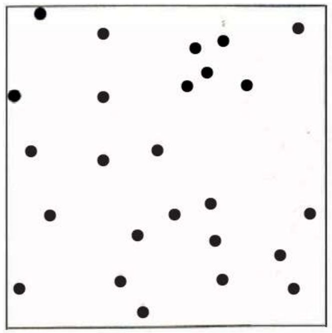
$f(x,y)=(1+\sin(x^2y^2))/2$,
$x,y\in[0,10.83]^2$, 512*512像素

随机采样(Random Sampling)



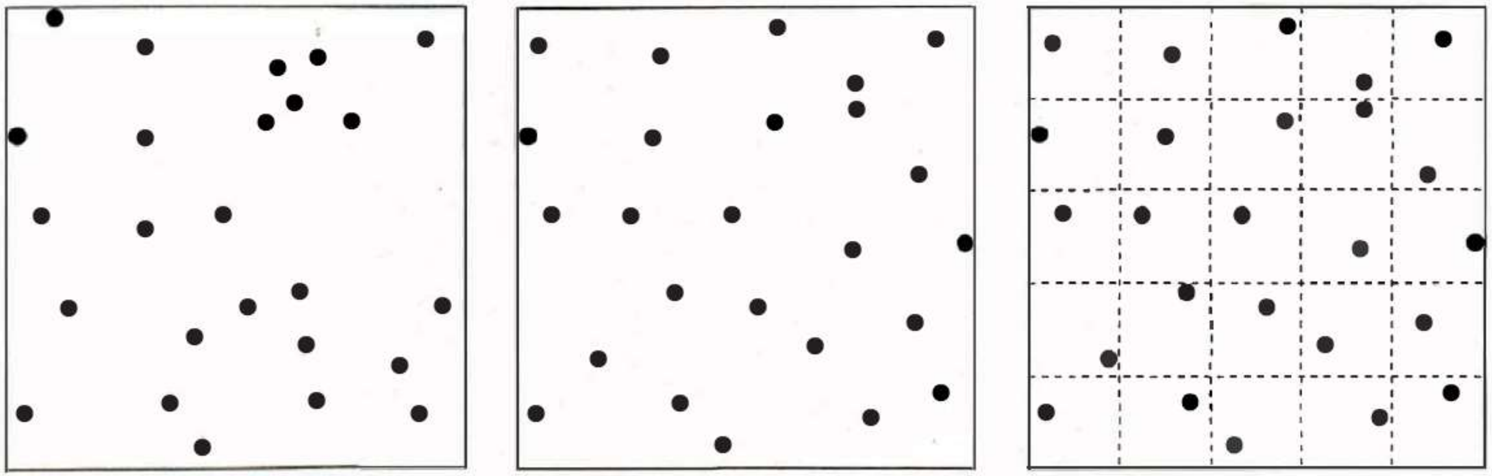
均匀采样引起的
moire pattern



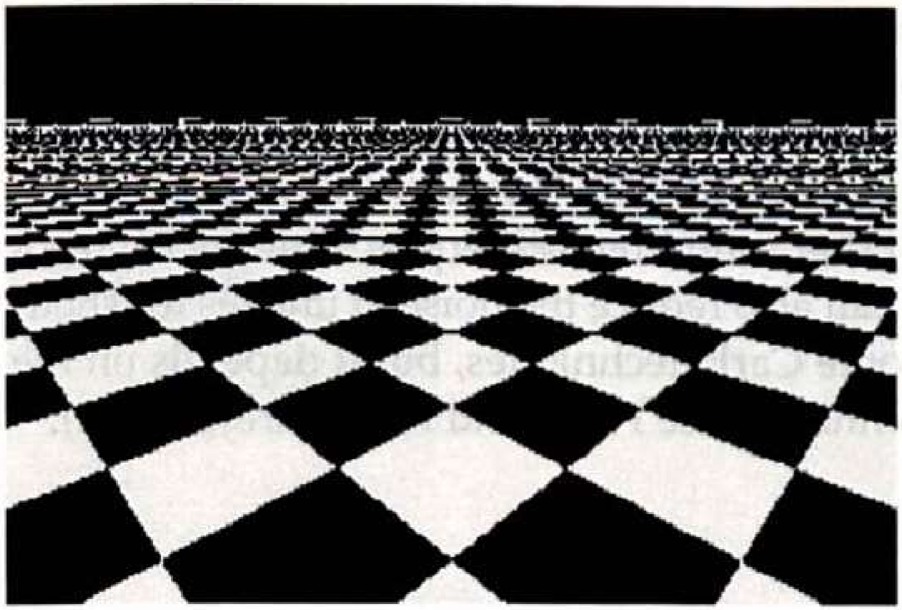
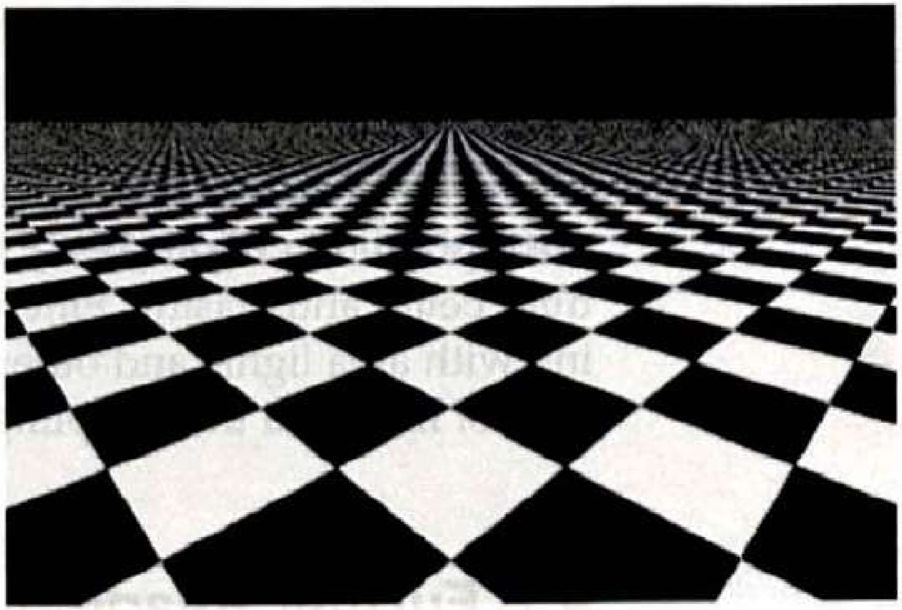


抖动采样(Jittered Sampling)

- 一方面保证像素采样的均匀分布，另一方面体现某种随机特征。



随机采样点分布　　　抖动采样点分布

计算题

- 求下图中以直线L为对称轴的对称变换矩阵**M**，其中L与y轴交于**B(0,*b*)**点，与x轴夹角为θ。