计算机图形学复习

一、基本内容

Ch1、计算机图形学简介

1.1 计算机图形学、图像处理、模式识别之间的关系

1.2 计算机图形标准：图形支撑软件标准、图形设备接口标准、图形数据交换标准

Ch2、图形基元的显示

2.1直线扫描转换算法：DDA算法、中点算法、Bresenham算法

2.2 圆的扫描转换算法：圆的八方向对称性；中点画圆法、Bresenham画圆法

2.3 区域填充：种子填充、多边形填充

2.4 区域：相连通的一组像素

2.5 连通分：四连通、八连通

2.6 种子点：选定区域内的一点，用以指定哪一个区域

Ch3、图形变换

3.1 齐次坐标[x y 1]或[x y z 1]、无穷远点[x y 0]或[x y z 0]

3.2 二维基本几何变换有：平移、比例、旋转、对称、错切

3.3 二维基本几何变换矩阵：

平移：$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_x & T_y & 1 \end{bmatrix}$ 比例：$\begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 旋转：$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$

错切：$\begin{bmatrix} 1 & 0 & 0 \\ b & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 和 $\begin{bmatrix} 1 & d & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

对称：$T_{my} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $T_{mx} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $T_{mO} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

$T_{m+45} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ $T_{m-45} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

3.4 三维基本几何变换有：平移、比例、旋转

3.5 三维基本几何变换矩阵：

平移：$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ T_x & T_y & T_z & 1 \end{bmatrix}$ 比例：$\begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

旋转：$\begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & \sin\theta & 0 \\ 0 & -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $\begin{bmatrix} \cos\theta & 0 & -\sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

3.6 二维和三维变换矩阵，组合变换矩阵

3.7 投影分类

$$\text{投影}\begin{cases} \text{平行投影}\begin{cases} \text{正交投影}\begin{cases} \text{正投影}\begin{cases} \text{正视投影} \\ \text{顶视投影} \\ \text{侧视投影} \end{cases} \\ \text{等轴投影} \end{cases} \\ \text{斜交投影} \end{cases} \\ \text{透视投影} \end{cases}$$

3.8 投影平面、投影中心、投影线、灭点、主灭点

3.9正投影矩阵

$$M_{正}=\begin{bmatrix}0&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}\qquad M_{顶}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&1\end{bmatrix}\qquad M_{侧}=\begin{bmatrix}1&0&0&0\\0&0&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$$

斜二测投影矩阵 $\begin{bmatrix}1&0&0&0\\0&1&0&0\\1/2\sqrt{2}&1/2\sqrt{2}&0&0\\0&0&0&1\end{bmatrix}$

一点透视矩阵 $\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&1/d\\0&0&0&0\end{bmatrix}$

3.10 直线段裁剪有：Cohen-Sutherland算法、中点分割算法、梁-Barsky算法

3.11 多边形的裁剪有Sutherland-Hodgman算法

Ch4、曲线和曲面

4.1 基本概念：型值点、插值、逼近

4.2 Hermite曲线和Hermite型曲面公式

$$P(t)=[t^3\ \ t^2\ \ t\ \ 1]\ H\begin{bmatrix} P_0 \\ P_1 \\ P_0' \\ P_1' \end{bmatrix} \qquad P(u,w)=[u^3\ u^2\ u\ 1]HMH^T\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}$$

4.3 Bezier曲线和曲面公式

$$P(t)=[t^3\ \ t^2\ \ t\ \ 1]\ B\begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \qquad P(u,w)=[u^3\ u^2\ u\ 1]BVB^T\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}$$

4.4 B样条曲线和曲面公式

$$P(t)=[t^3\ \ t^2\ \ t\ \ 1]\ Bs\begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \qquad P(u,w)=[u^3\ u^2\ u\ 1]BsVBs^T\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}$$

4.5 Hermite矩阵H，Bezier矩阵B，B样条矩阵Bs

$$H=\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad B=\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad Bs=\frac{1}{6}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}$$

4.6 三次Hermite曲线、Bezier曲线、B样条曲线在连接点处分别达到1阶、1阶、2阶导数连续

Ch5、图形运算

5.1 线段的交点计算：两线段求交

5.2 平面中的凸壳算法：Graham算法、Jarvis算法

5.3 包含算法：

简单多边形包含算法、折半查找凸多边形包含算法

Ch6、形体的表示及其数据结构

6.1 空间形体信息分图形信息和非图形信息，图形信息又分几何信息和拓扑信息

6.2 二维图形的边界表示：折线法、带树法

6.3 平面图形四叉树表示法的实现有：规则四叉树、线性四叉树、一对四式四叉树

6.4 三维几何模型的几何元素：点、边、环、面、体、体素

6.5 三维形体表示模型：线框模型、表面模型、实体模型

6.6 实体模型中的单链三表结构由三张表(顶点表、边表、面表)组成

6.7 常用实体表示法有构造实体几何法(CSG法)、边界表示法(Brep法)、特征表示法等

6.8 欧氏几何描述边界光滑的图形，分形几何描述边界极其毛糙的图形 (Koch曲线、Mandelbrot集、Julia集为分形图形)

Ch7、消除隐藏线和隐藏面的算法

7.1 消隐基本算法：客体空间算法、图像空间算法

7.2 消隐藏线算法有：凸多面体消隐算法、凹多面体消隐算法、浮动水平线算法

7.3 消隐藏面算法有：深度排序算法、 z缓冲算法、扫描线算法、区域分割算法、 BSP树算法、光线投射算法等

7.4 凸多面体消隐算法思想：根据凸多面体各个表面多边形的朝向确定该面是否可见，从而确定可见边

7.5 凹多面体消隐算法思想：对每一条朝视点面(可能可见面)的棱边投影求被其它朝视点面投影遮挡后的可见子线段

7.6 函数曲面网格的消隐可用浮动水平线算法

7.7 深度排序算法又叫画家算法

7.8 深度排序算法思想：依遮挡关系由远而近排序多边形，再依次画出各个面，后画面覆盖先画面

7.9 z缓冲算法思想：逐个扫描平面，近的面覆盖远的面，并用 z 缓冲器保留最近点的 z 值

7.10扫描线算法思想：依扫描线处理消隐，对扫描线上各个多边形依z值消隐

7.11 区域分割算法思想：不断分割区域，并画出简单遮挡关系的图形

7.12 BSP树算法思想：用大量平面不断将各多边形表面分成前后两类，然后从后往前依次画出各多边形表面

7.13 光线投射算法思想：由视点过像素的射线求与各表面交点，置像素色为最近交点(可见面上的点)颜色，如此确定所有像素颜色

Ch8、真实感图形的绘制

8.1 表面反射分漫反射和镜面反射

8.2 漫反射还分环境光漫反射和点光源漫反射

8.3 环境光反射模型： $I_1=I_aK_a$

8.4 点光源漫反射模型： $I_2=I_PK_d\cos\theta$

8.5 镜面反射模型： $I_3=I_PK_s\cos^n\alpha$

8.6 光照模型公式(局部)： $I=I_aK_a+\frac{I_P}{r+k}[K_d\cos\theta+K_s\cos^n\alpha]$

8.7 Gouraud明暗法又叫亮度插值明暗法，Phong 明暗法又叫法向量插值明暗法

8.8 表面纹理的实现有图案映射和法向量扰动

8.9 Whitted光照模型为整体模型，实现需用光线跟踪技术

8.10 加速光线跟踪算法的技术有场景分层次结合包围盒技术和空间分割技术

8.11 镜面反射、透射为主的环境适合使用光线跟踪方法

8.12 漫反射为主的环境适合使用辐射度方法

8.13 CIE色度图是国际照明委员会(CIE)制订的基于三原色原理的国际颜色标准

8.14 常用颜色模型:RGB模型、CMY模型、HSV模型、HLS模型

8.15 RGB模型属于加色系统， CMY模型属于减色系统

8.16 RGB模型、CMY模型是面向设备的颜色模型

8.17 HSV模型、HLS模型是面向用户的颜色模型

二、主要算法

1. **直线DDA算法**
2. **直线Bresenham算法**
3. **圆的Bresenham算法**
4. **以像素段为单位的种子填充算法**
5. **Cohen-Sutherland裁剪算法**
6. **梁-Barsky裁剪算法**
7. **Sutherland-Hodgmann多边形裁剪算法**
8. **Graham凸壳算法**
9. **Jarvis凸壳算法**
10. **简单多边形包含算法**
11. **凸多边形包含算法**
12. **折线法**
13. **带树法**
14. **凹多面体消隐算法**
15. **z-缓冲算法**
16. **区域分割算法**
17. **Gouraud明暗法**
18. **Phong明暗法**

1、DDA算法(数值微分法)

- $x_0 = x_A$, $y_0 = y_A$, $e = \max\{\ |y_B - y_A|\ ,\ |x_B - x_A|\ \}$
- 画点 ($\mathrm{int}(x_i+0.5)$, $\mathrm{int}(y_i+0.5)$)
- $x_{i+1} = x_i + (x_B - x_A) / e$ $\qquad$ $i=0,1,\ldots,e$
- $y_{i+1} = y_i + (y_B - y_A) / e$

2、Bresenham画线算法

- $(x_0 , y_0) = (x_A , y_A)$, $p_0=2\Delta y - \Delta x$
- 画点 (x_i , y_i)
- $x_{i+1} = x_i + 1$
- $y_{i+1} = \begin{cases} y_i + 1 & 若 p_i \geqslant 0 \\ y_i & \end{cases}$ $\qquad$ $i=0,1,\ldots,\Delta x$
- $p_{i+1} = \begin{cases} p_i + 2\Delta y - 2\Delta x & 若 p_i \geqslant 0 \\ p_i + 2\Delta y & \end{cases}$

3、圆的Bresenham算法：(上偏右)

- $x_0 = 0$, $y_0 = R$, $p_0=3-2R$

$$
\begin{cases}
\text{画点 } (x_i, y_i) \\
x_{i+1} = x_i + 1 \\
y_{i+1} = \begin{cases} y_i - 1 & \text{若} p_i \geqslant 0 \\ y_i \end{cases} \\
p_{i+1} = \begin{cases} p_i + 4(x_i-y_i)+10 & \text{若} p_i \geqslant 0 \\ p_i + 4x_i+6 \end{cases}
\end{cases}
\qquad i=0,1,2,\ \ldots\ldots\quad x_i \le y_i
$$

4、边界定义像素段种子填充算法： 种子点(x,y)

将种子点所在像素段入栈，执行如下步骤：

1. 取栈顶像素段进行填充
2. 从右至左检查所填像素段的上一横行，将找到的未填像素段入栈
3. 从右至左检查所填像素段的下一横行，将找到的未填像素段入栈
4. 若堆栈为空则算法结束，否则转 1) 执行

5、Cohen-Sutherland算法：直线段P_1P_2 ($P_1(x_1,y_1)$,$P_2(x_2,y_2)$)

1. **若Code(P_1)=0 且 Code(P_2)=0　　接受**
2. **若Code(P_1)& Code(P_2)≠0　　拒绝**
3. **若Code(P_1)≠0　　则令　P*=P_1**

   否则令 P*=P_2
4. **若Code(P*)&1000B≠0　与上边界求交点P'**

   否则若Code(P*)&0100B≠0与下边界求交点P'

   否则若Code(P*)&0010B≠0与右边界求交点P'

   否则　　与左边界求交点P'
5. **P' 代替 P*，转1**

6、梁-Barsky算法：直线段P_0P_1 ($P_0(x_0,y_0)$,$P_1(x_1,y_1)$)

令 $Q_L=-\Delta x$; $D_L=x_0-x_L$; $Q_r=\Delta x$; $D_r=x_r-x_0$

$Q_b=-\Delta y$; $D_b=y_0-y_b$; $Q_t=\Delta y$; $D_t=y_t-y_0$ $t_0=0$; $t_1=1$

对四条边界裁剪： $i=L , r , b , t$

1. 若 $Q_i=0$ 则若 $D_i < 0$ 拒绝
   若 $Q_i < 0$ 则 $t_0=\max\{ t_0 , D_i/Q_i \}$
   若 $Q_i > 0$ 则 $t_1=\min\{ t_1 , D_i/Q_i \}$
2. 若 $t_0 > t_1$ 则拒绝

7、Sutherland-Hodgman算法：

多边形顶点 $P_1, P_2, \ldots\ldots, P_n$，令 $P_{n+1}=P_1$

分别对四条边界执行下列步骤：

1. $S=P_1$, $P=P_2$
2. 循环执行 n 次
   ⑴ 若S，P在两侧，求交点I，I加入新顶点表
   ⑵ 若P在内侧，P加入新顶点表
   ⑶ $S=P$，$P=P_{i+1}$
3. 更新顶点表，更新n

8、Graham凸壳算法：

1. 倾角排序：取点集S中按 x , y 字典序最小的点O，
   其余与O连线按倾角排序 $\{Q_1,Q_2,\ldots,Q_n\}$
2. 准备扫描：v取Q_1
3. 扫描：循环执行下面操作直至下一点为Q_1
   判断：若连续三点v , v_1 , v_2 不形成左转，
   则删除v_1 点，v取v的前一点；
   否则v取v后一点

9、Jarvis凸壳算法：

1. 准备：v_0取点集S中x , y 字典序最小的点，
   v_0作为凸壳顶点，$S_1 \leftarrow S-\{v_0\}$， $d \leftarrow (0,-1)$， $u \leftarrow v_0$
2. 一步行进：求u到S_1中各点向量与d夹角最小点取为v_1
3. 准备下次：若$v_1 \neq v_0$，则执行
   v_1作为凸壳顶点， $S_1 \leftarrow S-\{u,v_1\}$，
   $d \leftarrow$向量uv_1，$u \leftarrow v_1$，转2

10、简单多边形包含算法:点$P(x_P,y_P)$ 多边形$P_0P_1...P_{n-1}$ ($P_i(x_i,y_i)$)

1. 准备：$(x_n,y_n) \leftarrow (x_0,y_0)$ ， $m \leftarrow -1$
2. 对每一条边P_iP_{i+1} 反复进行下列操作
   - (i) 排除必不相交情形：若下列有一条件成立，则转2
     - ① 边在P向下射线右且不碰线
     - ② 边在P向下射线左，可碰线
     - ③ 边完全在点P上
   - (ii) 计算：求出边与经过P的垂直线交点(x_P,y')，判断
     - ① 若$y'=y_P$，则点在多边形边界上，算法结束
     - ② 若$y'<y_P$，则 $m \leftarrow -m$
3. 判断：若$m= -1$ 则P点在多边形外部；否则在内部

11、折半查找凸多边形包含算法：　　凸多边形$P_0P_1...P_{n-1}$

1. **准备：$i \leftarrow 1$，$j \leftarrow n-1$**
2. **查找是否结束：若 $j-i=1$ 则转4**
3. **折半查找：$k \leftarrow int((i+j)/2)$，判别P与$P_0P_k$关系**
   - **① 若P与P_0P_k共线，若P在线段P_0P_k上，则P在内，否则P在外，算法结束**
   - **② 若P在左侧，则 $i \leftarrow k$，否则 $j \leftarrow k$，转2**
4. **最后检查：若P在$\Delta P_0P_iP_j$内，则P在内，否则P在外**

12、折线法算法： 离散点$P_0P_1...P_n$ (每次检查k_0+1个点)

1. 取P_0点, 对P_0, ... ,P_{k0} 共线性检查分解求P_0P_k作为线段L_1
2. 反复取P_k点开始的k_0+1个点(记为P_j,...,P_k)执行下列操作
   ① 共线性检查分解求P_jP_k作为线段L_2
   ② 向量夹角检查合并：
   若L_1和L_2夹角不够小，则取P_j点，L_1取L_2
   若L_1和L_2夹角足够小，则合并L_1和L_2包含的点作为P_j,...,P_k，共线性检查分解得P_jP_k作为L_1
3. 取最后点P_n

13、带树法算法： 离散点$P_0P_1...P_n$ (分辨率w_0)

1. 找出P_0P_n确定的矩形带段，用节点记录并作为根节点
2. 找出矩形带段P_iP_j中离连线P_iP_j最远的点P_k，用节点记录矩形带段P_iP_k和P_kP_j，分别作为节点P_iP_j的左右子节点
3. 反复执行2直到节点记录的矩形带段的$w_L+w_r \le w_0$，该类节点作为叶节点

14、 凹多面体消隐算法：

1. 用外法线求得可能可见面
2. 对每一条可能可见棱边E_i，执行(i) ~ (iii)

   (i) 对每一个可能可见面执行 (a) ~ (c)

   (a) 包围盒检测，不遮挡E_i则转 (i)

   (b) 深度检测，不遮挡E_i则转 (i)

   (c) 平面与E_i投影求交，保存不可见子线段

   (ii) 棱边E_i投影的各不可见子线段求并，并求出可见子线段

   (iii) 画出E_i的可见子线段

15、 z缓冲算法：

1. z缓冲器取最大值，屏幕像素取背景色
2. 逐个扫描多边形，执行
   对多边形每个点，求z值，并与z缓冲器中对应z比较，
   若点更近则替换z缓冲器值和相应像素颜色

16、区域分割算法：

1. 初始化区域为全屏幕
2. 根据区域创建可见多边形表，按z_{min}升序排序
3. 可见性判断
   (a) 表为空，区域取背景色
   (b) 表中只有一个多边形，则多边形着色，其余取背景色
   (c) 表中只有一个包围多边形，取多边形色
   (d) 区域缩为一点(x,y)，计算并取最小z值的多边形颜色
4. (a)~(d)均不满足，则分割为4个子区域，对每个子区域转2

17、Gouraud明暗：

1. 计算各多边形平面的单位法向量
2. 对拥有同一顶点的各多边形平面的法向量取平均值，得顶点法向量
3. 用光照模型求出各顶点的亮度值
4. 对每条边用两顶点的亮度插值，再沿每一条扫描线在各边之间作亮度插值，从而实现多边形的明暗处理

18、Phong明暗法：

1. 计算各多边形平面的单位法向量
2. 对拥有同一顶点的各多边形平面的法向量取平均值，得顶点法向量
3. 对每条边用两顶点的单位法向量插值，再沿每一条扫描线在各边之间作单位法向量的插值，求出扫描线各点的法向量
4. 利用法向量求出亮度值

三、计算题复习

计算1：用圆的Bresenham算法计算R=11的上偏右圆弧像素坐标

解：

i	(x_i, y_i)	p_i
0	(0,11)	-19
1	(1,11)	-13
2	(2,11)	-3
3	(3,11)	+11
4	(4,10)	-11
5	(5,10)	+11
6	(6,9)	+1
7	(7,8)	-1
8	(8,8)	+33
9	(9,7)	+43

- $x_0 = 0$, $y_0 = R$, $p_0=3-2R$
- 画点 (x_i , y_i)
- $x_{i+1} = x_i + 1$
- $y_{i+1} = \begin{cases} y_i - 1 & p_i \geqslant 0 \\ y_i & \end{cases}$
- $p_{i+1} = \begin{cases} p_i + 4(x_i-y_i)+10 & p_i \geqslant 0 \\ p_i + 4x_i+6 & \end{cases}$

计算2：求A(0,0)B(2,2)C(0,2)关于直线y=0.75x
对称变换后的坐标

解：变换矩阵为：$T=R(-\theta)T_{mx}R(\theta)$，其中： $\tan\theta=3/4$
易知 $\cos\theta=0.8$， $\sin\theta=0.6$，故

$$T=R(-\theta)T_{mx}R(\theta)=\begin{bmatrix} 0.8 & -0.6 & 0 \\ 0.6 & 0.8 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0.8 & 0.6 & 0 \\ -0.6 & 0.8 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$=\begin{bmatrix} 0.28 & 0.96 & 0 \\ 0.96 & -0.28 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

计算A'B'C'坐标：

$$\begin{array}{c} A \\ B \\ C \end{array}\begin{bmatrix} 0 & 0 & 1 \\ 2 & 2 & 1 \\ 0 & 2 & 1 \end{bmatrix}\begin{bmatrix} 0.28 & 0.96 & 0 \\ 0.96 & -0.28 & 0 \\ 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} 0 & 0 & 1 \\ 2.48 & 1.36 & 1 \\ 1.92 & -0.56 & 1 \end{bmatrix}\begin{array}{c} A' \\ B' \\ C' \end{array}$$

故A'(0,0),B'(2.48,1.36),C'(1.92,-0.56)

计算3：求窗口(1,2)-(6,6)到视区(0,50)-(500,450)的视见变换矩阵H及P(2,3)变换到视区的点P'的坐标。

解： $H=T(-1,-2)\cdot S(100,100)\cdot T(0,50)=\begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ -100 & -150 & 1 \end{bmatrix}$

$P[\,2\ \ 3\ \ 1\,]\cdot H = [100\ \ 150\ \ 1]\ P'$

计算4：由端点$P_0(1,1)$, $P_1(5,1)$及端点切向量$P_0'(0,1)$, $P_1'(0,-1)$,
求Hermite曲线上点P(0.1),P(0.5)的坐标。

解：

$$P(t)=[t^3\ \ t^2\ \ t\ \ 1]\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 5 & 1 \\ 0 & 1 \\ 0 & -1 \end{bmatrix}=[t^3\ \ t^2\ \ t\ \ 1]\begin{bmatrix} -8 & 0 \\ 12 & -1 \\ 0 & 1 \\ 1 & 1 \end{bmatrix},\ t\in[0,1]$$

故：P(0.1)=(1.112,1.09), P(0.5)=(3,1.25)

计算5：给定点$P_0(0,1)$, $P_1(2,3)$, $P_2(4,0)$, $P_3(5,2)$ ， 求Bezier曲线上点P(0.1), P(0.5)的坐标。

解：

$$P(t)=[t^3\ \ t^2\ \ t\ \ 1]\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 2 & 3 \\ 4 & 0 \\ 5 & 2 \end{bmatrix}=[t^3\ \ t^2\ \ t\ \ 1]\begin{bmatrix} -1 & 10 \\ 0 & -15 \\ 6 & 6 \\ 0 & 1 \end{bmatrix},\ t\in[0,1]$$

故：P(0.1)=(0.599,1.46), P(0.5)=(2.875,1.5)

计算6：已知：

$V_{00}(0,0,-2)$ $V_{01}(0,1,2)$ $V_{02}(0,2,0)$ $V_{03}(0,3,4)$

$V_{10}(1,0,0)$ $V_{11}(1,1,1)$ $V_{12}(1,2,2)$ $V_{13}(1,3,3)$

$V_{20}(2,0,2)$ $V_{21}(2,1,0)$ $V_{22}(2,2,4)$ $V_{23}(2,3,2)$

$V_{30}(3,0,4)$ $V_{31}(3,1,-1)$ $V_{32}(3,2,6)$ $V_{33}(3,3,1)$

求双三次B样条曲面的x(u,w) ,y(u,w),z(u,w)。

解： 由条件知V矩阵如下：

$$V_x=\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \end{bmatrix} \quad V_y=\begin{bmatrix} 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \end{bmatrix} \quad V_z=\begin{bmatrix} -2 & 2 & 0 & 4 \\ 0 & 1 & 2 & 3 \\ 2 & 0 & 4 & 2 \\ 4 & -1 & 6 & 1 \end{bmatrix}$$

故：$x(u,w)=[u^3\ u^2\ u\ 1]\ B_S\ V_x\ B_S^T\ [w^3\ \ w^2\ \ w\ \ 1]^T$

$$=[u^3\ u^2\ u\ 1]\ \frac{1}{6}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 2 \\ 3 & 3 & 3 & 3 \end{bmatrix}\frac{1}{6}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 0 & 4 \\ -3 & 3 & 3 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}$$

$$=[u^3\ u^2\ u\ 1]\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix} =u+1$$

$$y(u,w)=[u^3\ u^2\ u\ 1]\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix} =w+1$$

$$z(u,w)=[u^3\ u^2\ u\ 1]\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -2 & 3 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}\begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix} =-2uw^3+3uw^2+w+1$$

计算7：求三角形ABC的边界与线段P_0P_1的交点
(其中A(2,2)B(8,5)C(4,8)　$P_0(0,0)P_1(5,7)$)

解：P_0P_1方程： $\begin{cases} x=5t \\ y=7t \end{cases}$

与AB: $\begin{cases} x=2+6\lambda \\ y=2+3\lambda \end{cases}$ 求交 $\begin{cases} 5t=2+6\lambda \\ 7t=2+3\lambda \end{cases}$ 解得: $\begin{cases} t=6/27 \\ \lambda= -4/27<0 \end{cases}$ 不相交

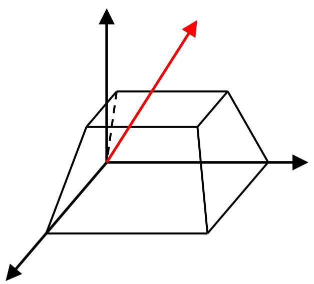
与BC: $\begin{cases} x=8-4\mu \\ y=5+3\mu \end{cases}$ 求交 $\begin{cases} 5t=8-4\mu \\ 7t=5+3\mu \end{cases}$ 解得: $\begin{cases} t=44/43>1 \\ \mu= 31/43 \end{cases}$ 不相交

与AC: $\begin{cases} x=2+2\gamma \\ y=2+6\gamma \end{cases}$ 求交 $\begin{cases} 5t=2+2\gamma \\ 7t=2+6\gamma \end{cases}$ 解得: $\begin{cases} t=1/2 \\ \gamma= 1/4 \end{cases}$ 相交

交点(2.5,3.5)

计算8：已知四棱台ABCD-A'B'C'D’和视线向量s(1,1,2)，求可见面，其中A(1,1,2)B(3,1,2)C(3,3,2)D(1,3,2)A'(0,0,0)B'(4,0,0)C'(4,4,0)D'(0,4,0)。

解：



ABC面：$\begin{vmatrix} 1 & 1 & 2 & 0 \\ 1 & 1 & 2 & 1 \\ 3 & 1 & 2 & 1 \\ 3 & 3 & 2 & 1 \end{vmatrix} = 8 > 0$ 可见

C'B'A'面：$\begin{vmatrix} 1 & 1 & 2 & 0 \\ 4 & 4 & 0 & 1 \\ 4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{vmatrix} = -32 < 0$ 不可见

BAA'面：$\begin{vmatrix} 1 & 1 & 2 & 0 \\ 3 & 1 & 2 & 1 \\ 1 & 1 & 2 & 1 \\ 0 & 0 & 0 & 1 \end{vmatrix} = 0$ 不可见

A'AD面：$\begin{vmatrix} 1 & 1 & 2 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 3 & 2 & 1 \end{vmatrix} = 0$ 不可见

DCC'面：$\begin{vmatrix} 1 & 1 & 2 & 0 \\ 1 & 3 & 2 & 1 \\ 3 & 3 & 2 & 1 \\ 4 & 4 & 0 & 1 \end{vmatrix} = 8 > 0$ 可见

C'CB面：$\begin{vmatrix} 1 & 1 & 2 & 0 \\ 4 & 4 & 0 & 1 \\ 3 & 3 & 2 & 1 \\ 3 & 1 & 2 & 1 \end{vmatrix} = 8 > 0$ 可见

所以可见面为：ABC面，DCC'面，C'CB面。