

2018-2019年度第二学期 00106501

计算机图形学



童伟华 管理科研楼1205室

E-mail: tongwh@ustc.edu.cn

中国科学技术大学 数学科学学院

<http://math.ustc.edu.cn/>





第二节 隐藏面消除

流水线中的位置

- 在顶点经过几何变换后，其所定义的几何对象经过了装配与裁剪
 - 得到的是一组点、线段和多边形
 - 投影后，所有的图元都有可能出现在显示设备上
- 此时需要进行隐藏面消除
 - 有多种不同的解决方案

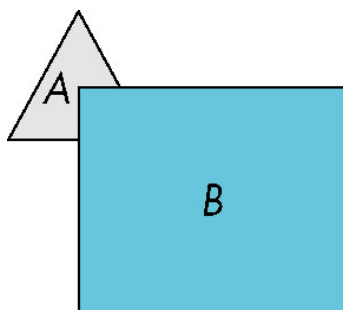
对象空间的算法



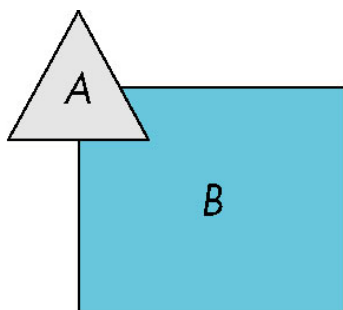
■ 考虑由 k 个三维不透明多边形构成的场景

- 每个多边形认为是单独的一个对象

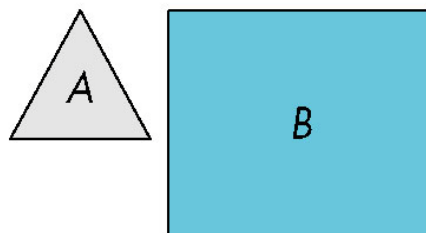
■ 两两考虑对象，检测相互之间的位置



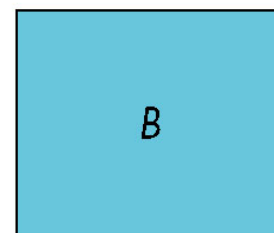
(a)



(b)



(c)



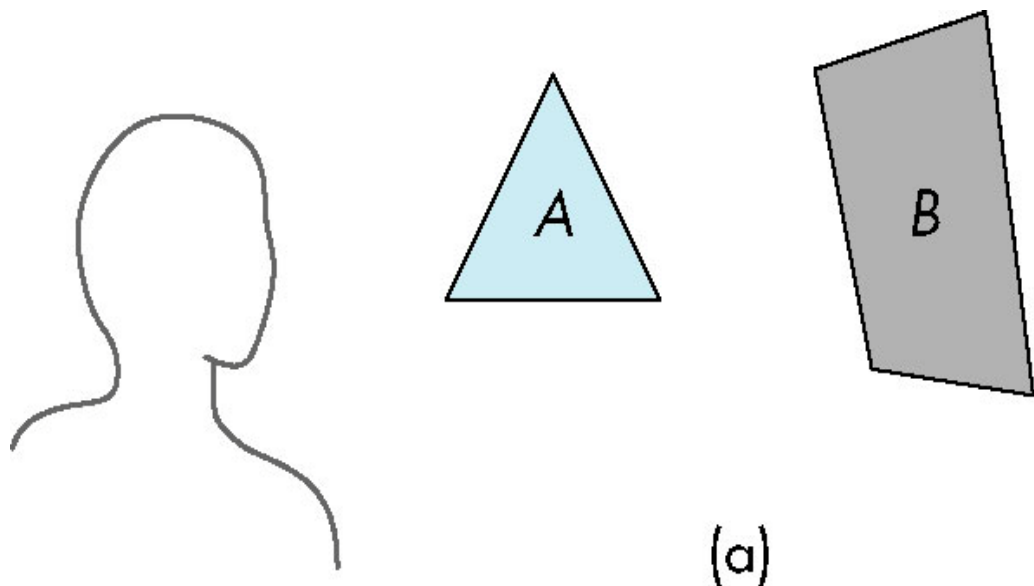
(d)

■ 最糟糕的情形： n 个多边形复杂度为 $O(n^2)$

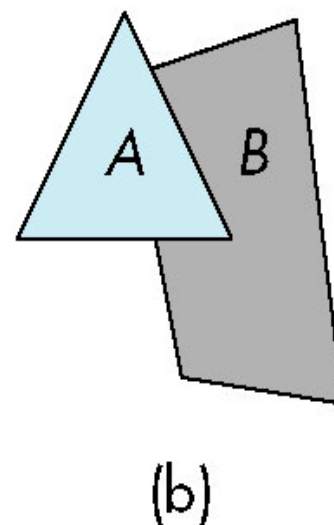
画家算法



- 把多边形沿从后到前的顺序显示出来，从而被遮住的多边形在显示时被重新激活



从观察者的角度来说，B在A后面



先画B，再画A

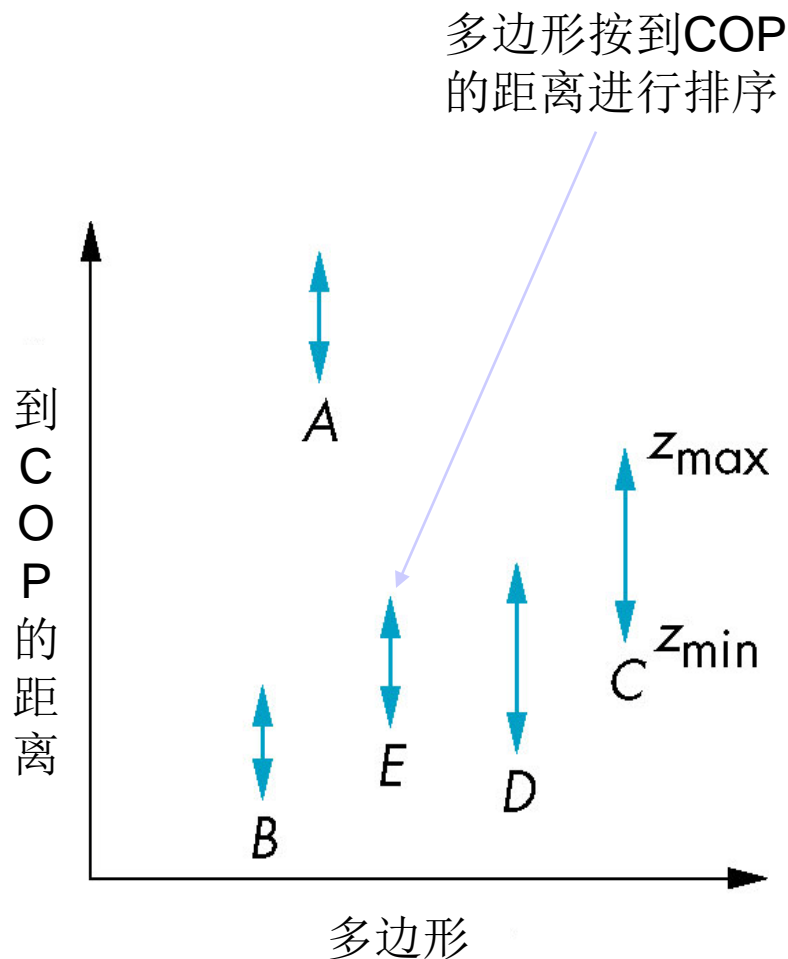
深度排序



■ 需要首先对多边形进行排序

- 排序的复杂度为 $O(n \log n)$
- 并不是所有的多边形都完全在其它多边形的前面或后面

■ 进行排序时，先处理简单情形，再处理困难情形



简单情形

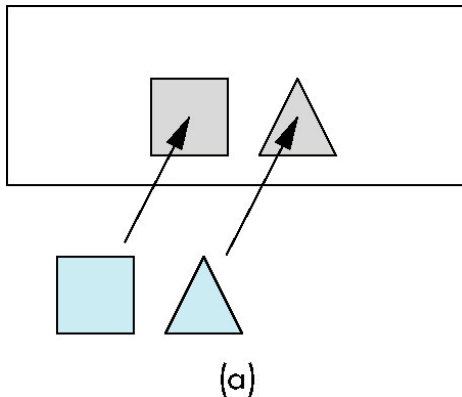
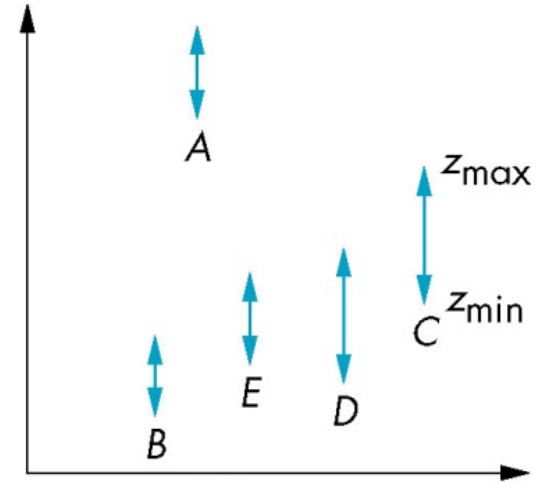


■ 多边形A位于所有其它多边形后面

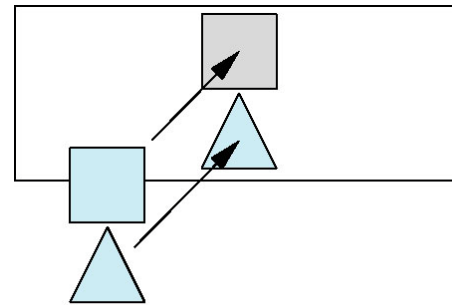
- 可以绘制出来

■ 多边形在Z方向有重叠，但在X或Y方向没有重叠

- 可以分别显示绘制出来

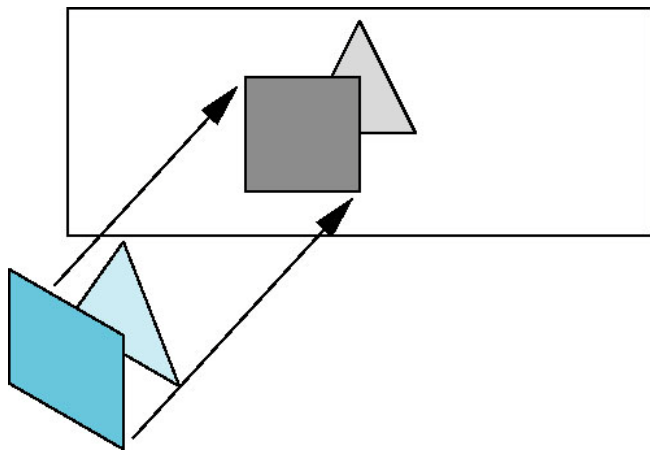


(a)

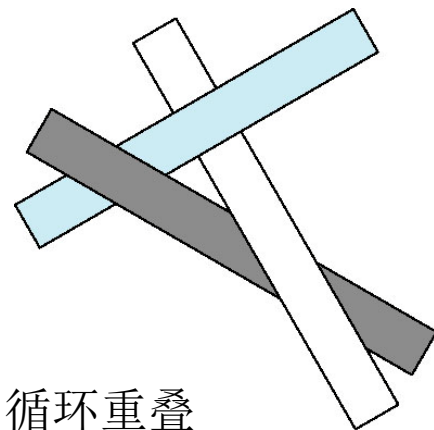


(b)

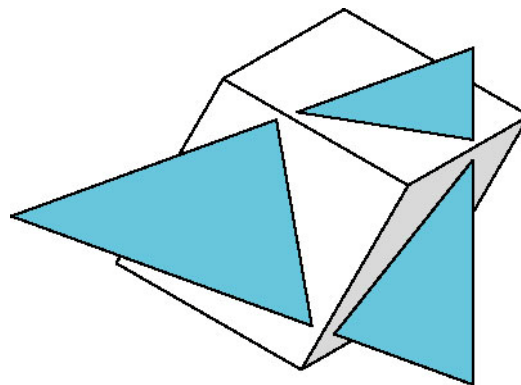
困难情形



在所有方向都有重叠，
但其中一个完全在另
一个的一侧



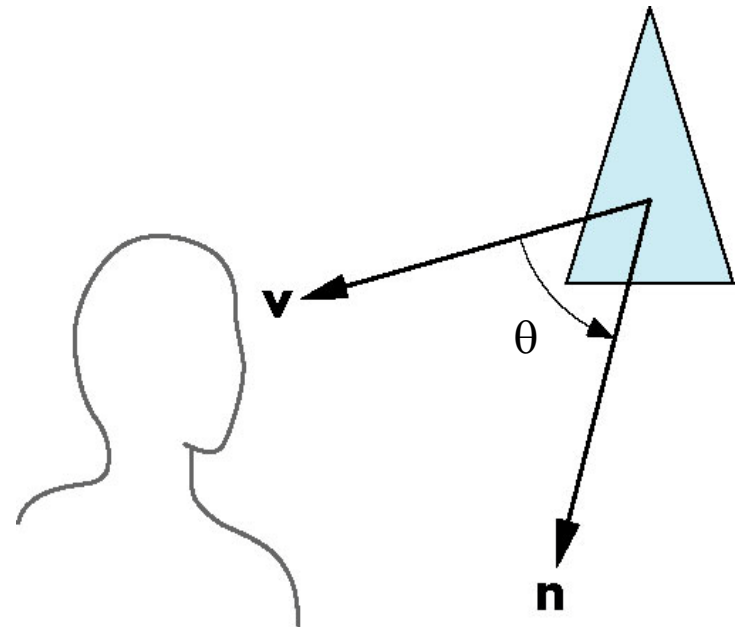
循环重叠



贯穿

背面剔除 (back-face removal)

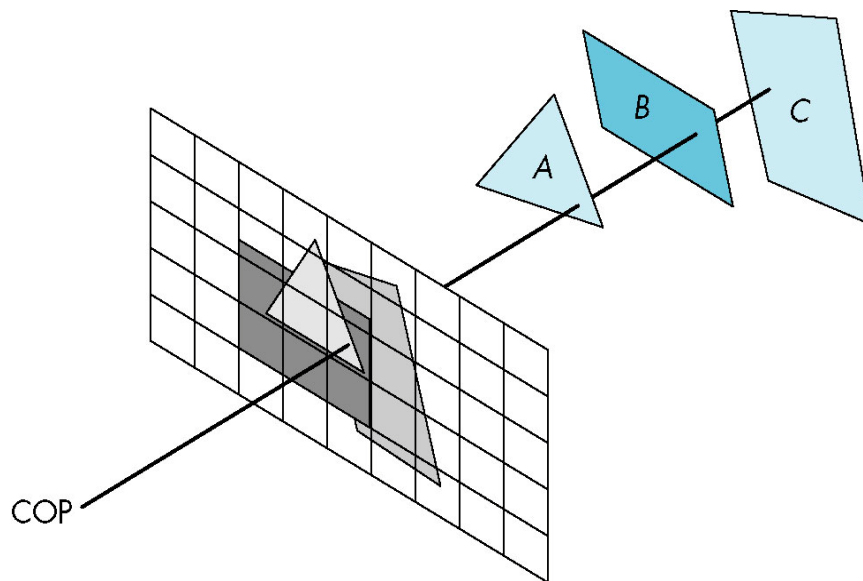
- 面是可见的, 如果 $-90^\circ \leq \theta \leq 90^\circ$
 - 等价于 $\cos \theta \geq 0$ 或者 $v \cdot n \geq 0$
- 平面具有形式 $ax + by + cz + d = 0$, 但对于规范化视线向量 $n = (0, 0, 1, 0)$
 - 只需要检测 c 的符号
 - 在OpenGL中可以激活背面剔除功能, 但是如果具有非凸对象, 结果可能不正确



图像空间的过程

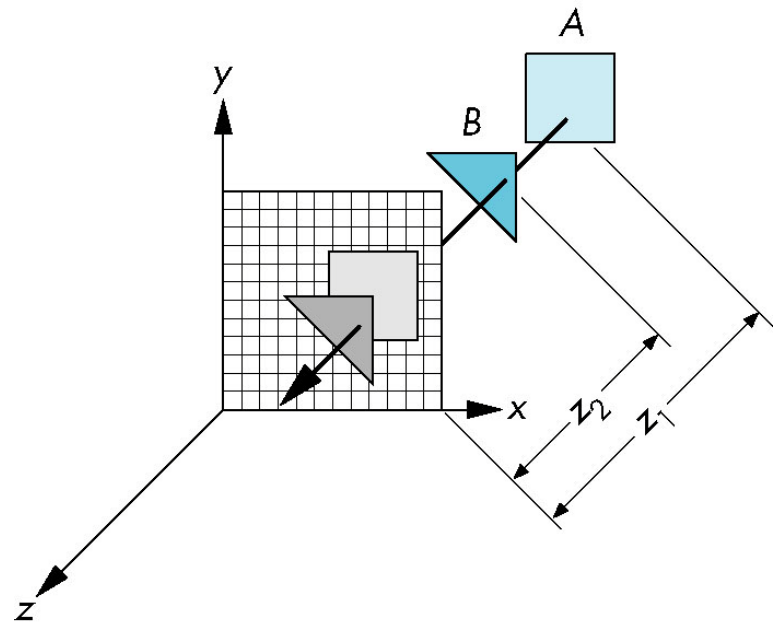


- 对每条投影线（对于 $n \times m$ 分辨率的帧缓冲区，共有 nm 条投影线），找到 k 个多边形中最近的那个
- 复杂度 $O(nmk)$
- 光线跟踪中用到
- Z缓冲区算法



Z缓冲区算法

- 应用一个称为Z缓冲区或者深度缓冲区的地方存储在每个像素，到目前为止找到的最近对象的深度
- 但显示每个多边形时，把它的深度与Z缓冲区中存储的深度进行比较
- 如果新值小的话，把新的亮度值放到颜色缓冲区中并且用新深度更新Z缓冲区

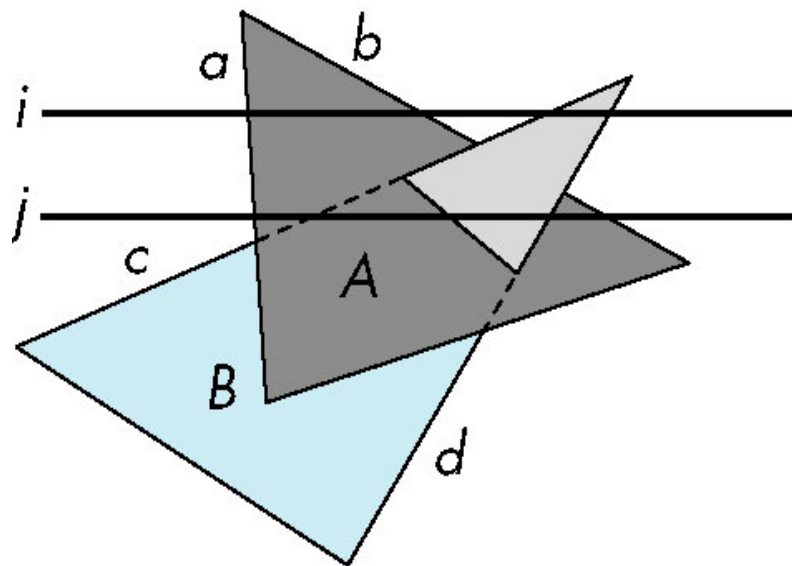


扫描线算法



■ 在扫描线算法中把明暗处理算法与隐藏面消除算法结合在一起

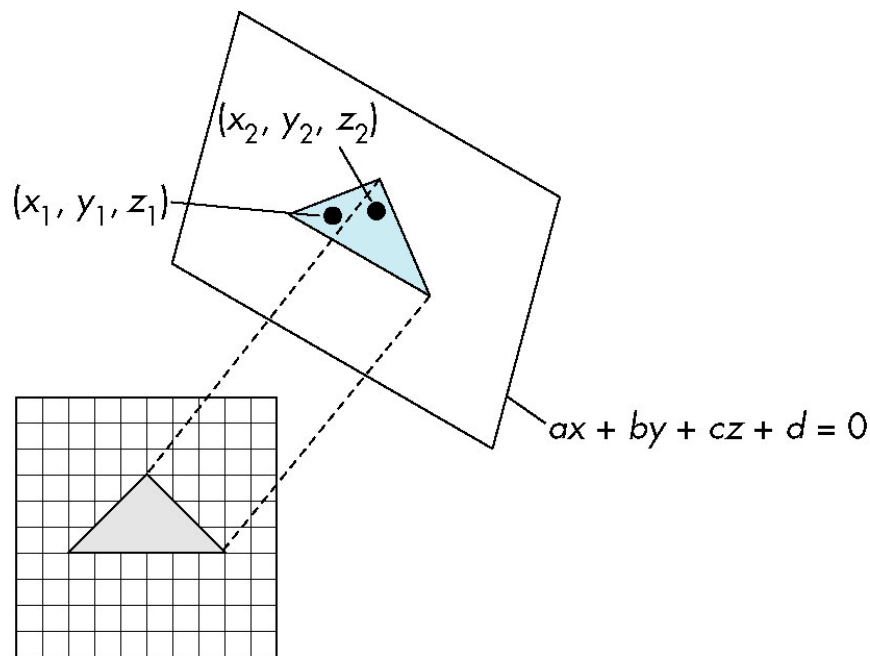
- 扫描线*i*: 不需要深度信息, 适合于没有多边形或者只有一个多边形
- 扫描线*j*: 只有当遇到多个多边形时, 才需要深度信息



效率的提高



- 如果逐条扫描线遍历，从左向右移动一条扫描线，那么深度的改变满足 $a \Delta x + b \Delta y + c \Delta z = 0$
- 沿每条扫描线 $\Delta y = 0$,
 $\Delta z = -a/c \Delta x$
- 在屏幕空间上 $\Delta x = 1$



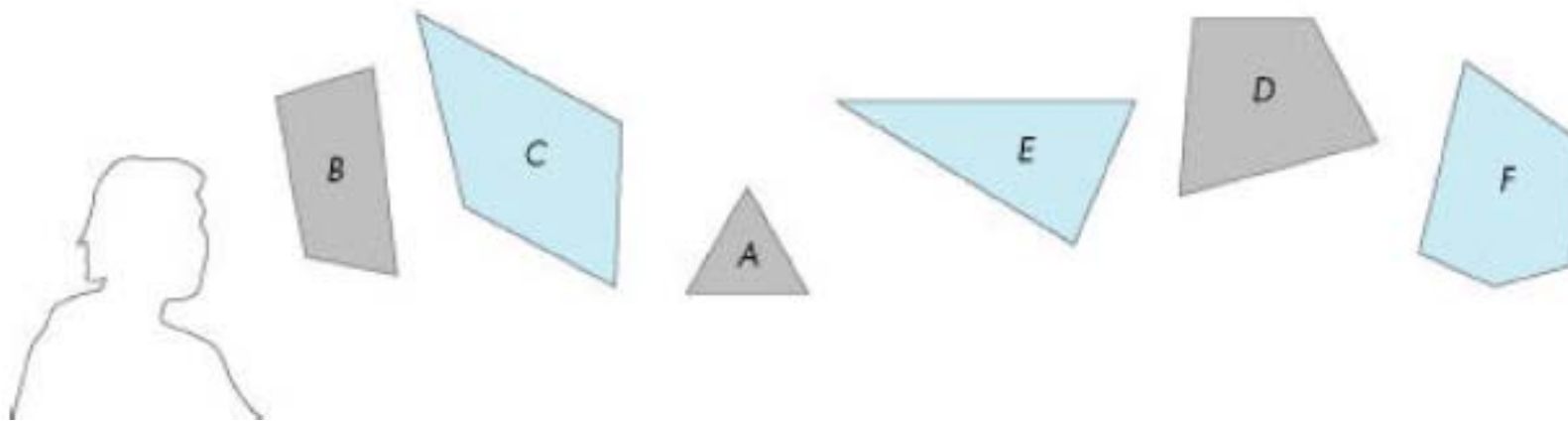
■ 需要相应的数据结构存储下述信息

- 每个多边形的标志(内部或外部)
- 扫描线的增量结构, 存储遇到的边的信息
- 平面的参数

可见面检测



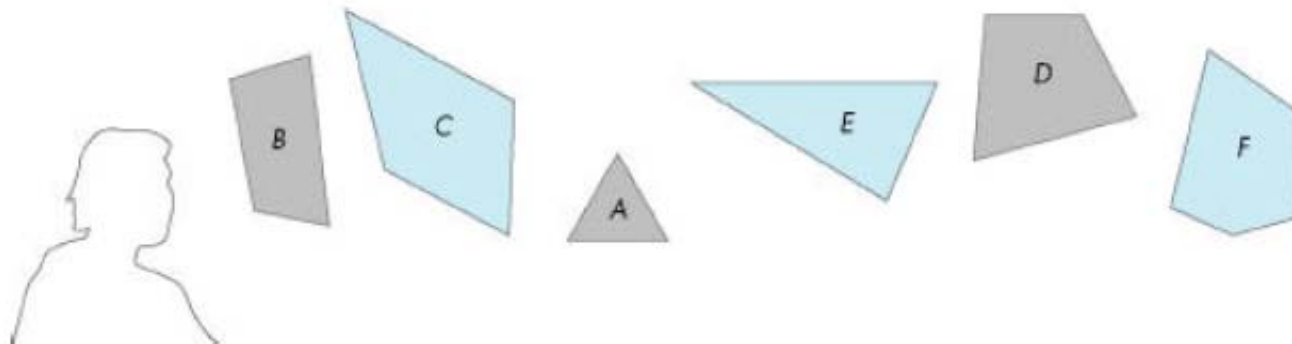
- 在许多实时应用系统中（例如游戏），我们希望能在应用程序中消除尽可能多的不可见对象
 - 从而降低流水线体系的负担
 - 降低出现数据通信阻塞的可能
- 利用二进空间剖分（binary spatial partition, BSP）树结构剖分空间



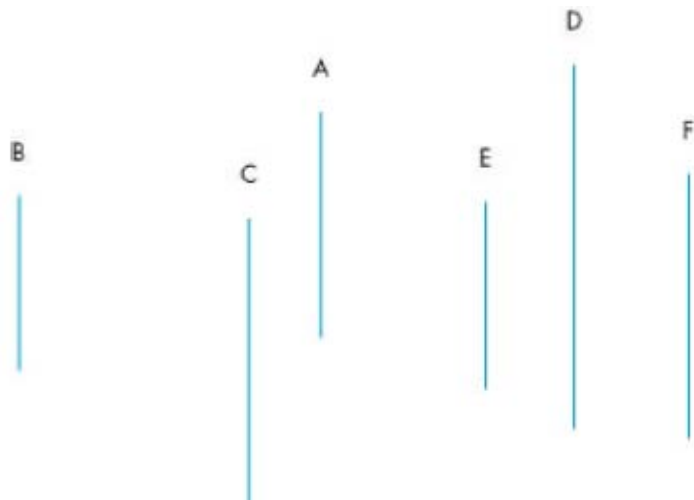
简单示例



考虑六个
平行多边形



平面B和C位于平面A
的一侧，平面D，
E和F位于平面A的
另一侧



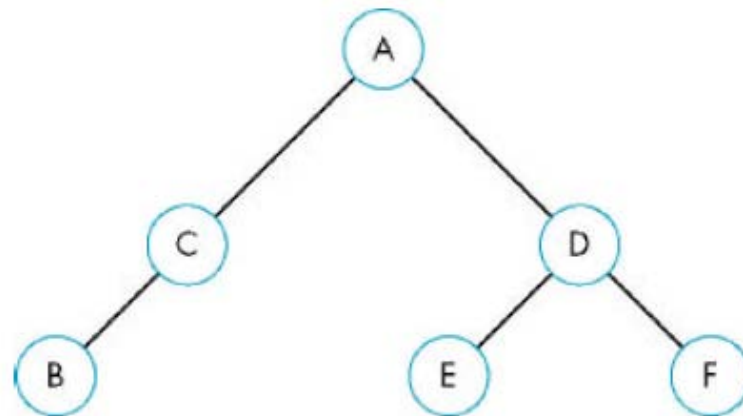
顶视图

■ 可以连续进行剖分

- 平面C把B与A分开
- 平面D把E与F分开

■ 可以把这些信息放在一个BSP树中

- 用于可见性与遮挡检测



Thanks for your attention!

