

# Fundamentos de Processamento Gráfico

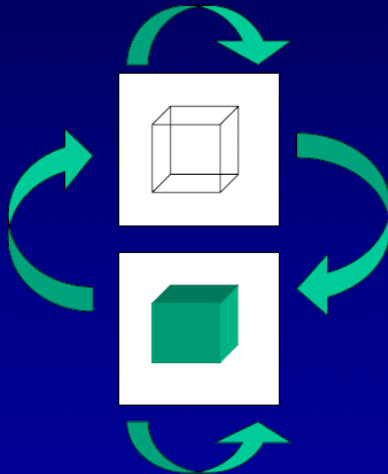
Helton H. Biscoaro ; Fátima Nunes

2 de maio de 2022

# Áreas Correlatas

Modelagem Geométrica

Visão  
Computacional



Computação  
Gráfica

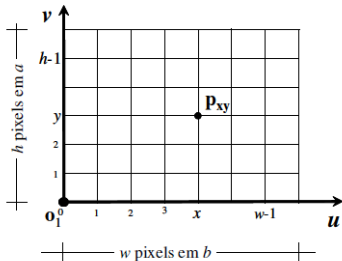
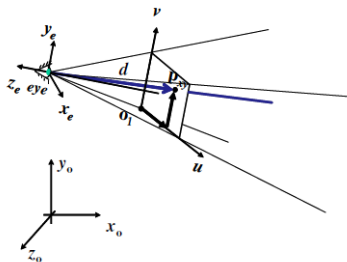
Processamento de Imagens

# Conceitos usados nesta aula

- Geometria Euclidiana;
- Espaços vetoriais;
- Transformações lineares;
- Operações (soma, multiplicação por escalar ) vetoriais;
- Sistemas de coordenadas;
- Operações com conjuntos;

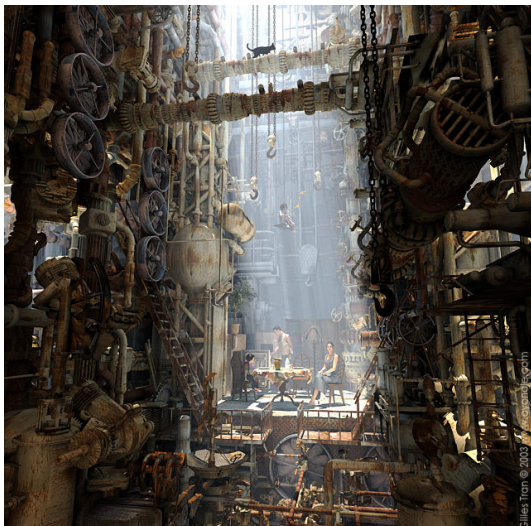
# Pipeline de Visualização

Projeção de um ponto.



# Pipeline de Visualização

Uma cena um pouquinho mais complexa.

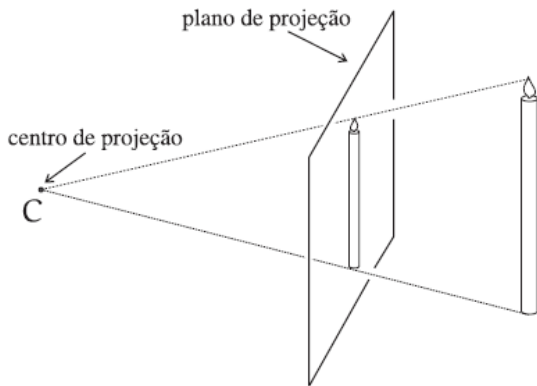


# Pipeline de Visualização

- **Espaço do Objeto:** Onde cada objeto é modelado. Ele possui um sistema de coordenadas associado à geometria do objeto.
- **Espaço de Cena:** É um sistema de coordenadas global
- **Espaço de Câmera:** Esse espaço é determinado pelo sistema de coordenadas associado à projeção cônica da câmera virtual
- **Espaço Normalizado:** Espaço utilizado para operações de recorte de objetos que estão fora do campo de visão da câmera
- **Espaço de Ordenação:** Espaço que facilita a operação de visibilidade (verifica se um objeto está ou não na frente de outro)
- **Espaço de Imagem:** Espaço da tela virtual no plano de projeção da câmera virtual
- **Espaço do Dispositivo:** (Espaço de Tela) Espaço associado à superfície de exibição do dispositivo de saída gráfica

# Pipeline de Visualização Espaço de Cena:

## Definição da Câmera Virtual



## Pipeline de Visualização Espaço de Câmera:

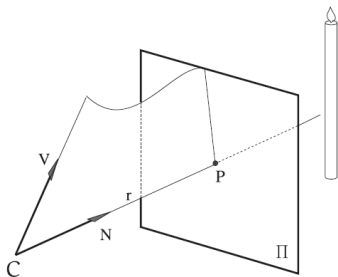
Dados:

Centro ótico  $C$ ;

Ponto de visão  $P$ ;

Vetor Vertical  $V$ ;

Precisamos definir um sistema de coordenadas ortonormais com centro em  $C$ .





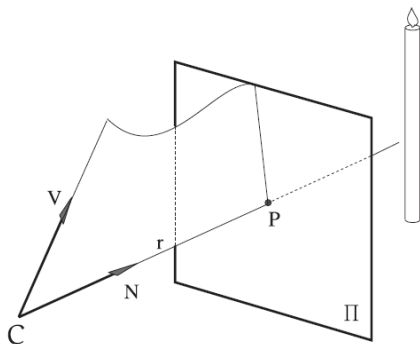
# Pipeline de Visualização Espaço de Câmera:

Definimos

$$\text{O vetor } n = \frac{P-C}{\|P-C\|};$$

$$\text{O vetor } v = \frac{V - \langle V, n \rangle n}{\|V - \langle V, n \rangle n\|};$$

$$\text{O vetor } u = v \times n.$$



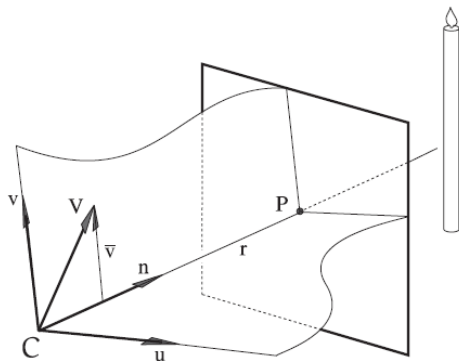
# Pipeline de Visualização Espaço de Câmera:

Definimos

$$\text{O vetor } n = \frac{P-C}{\|P-C\|};$$

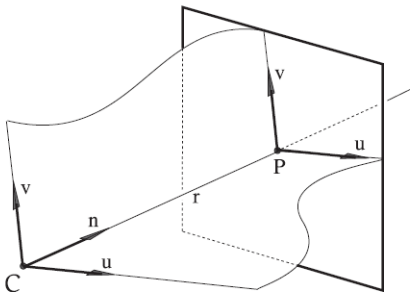
$$\text{O vetor } v = \frac{V - \langle V, n \rangle n}{\|V - \langle V, n \rangle n\|};$$

$$\text{O vetor } u = v \times n.$$



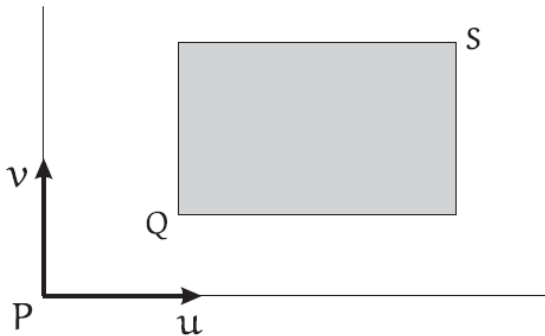
## Pipeline de Visualização Espaço de Imagem:

O Espaço de Câmera define naturalmente uma sistema de coordenadas no plano de projeção.



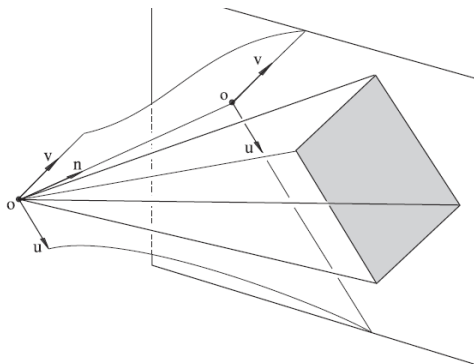
## Pipeline de Visualização Espaço de Imagem:

O Espaço de Câmera define naturalmente uma sistema de coordenadas no plano de projeção.



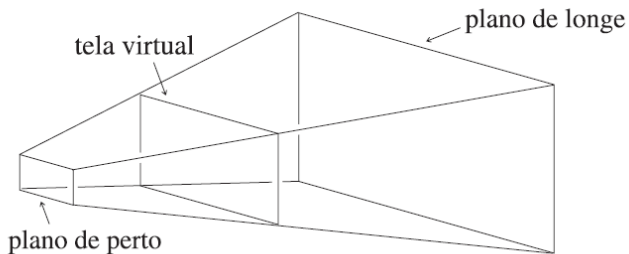
# Pipeline de Visualização Espaço de Imagem:

Pirâmide de Visão.



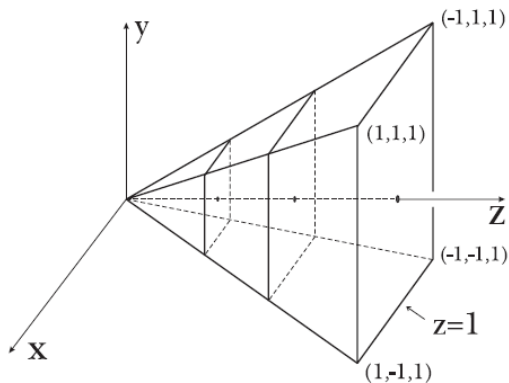
## Pipeline de Visualização Espaço de Imagem:

Volume de Visão:  
Operações de recorte e de ordenação.



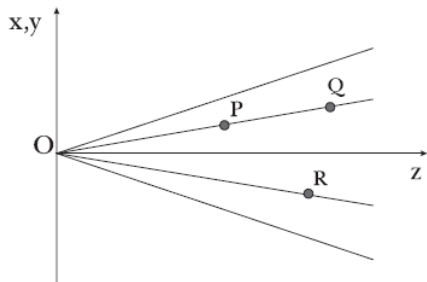
## Pipeline de Visualização Espaço de Normalizado:

Volume de Visão:  
Operações de recorte.

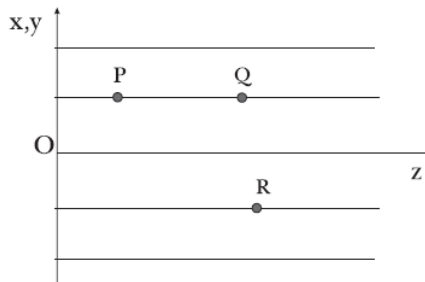


# Pipeline de Visualização Espaço de Ordenação:

Volume de Visão:  
Operações ordenação.



(a)



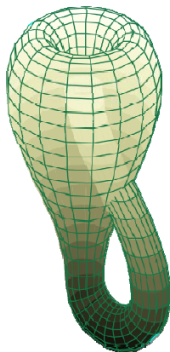
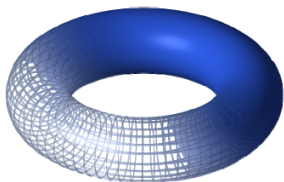
(b)



A coleção de métodos para descrever a forma e outras características geométricas de um objeto é conhecida como **Modelagem Geométrica**

# Modelagem

A geometria pode ser complicada.



# Modelagem Histórico:

- Modelagem por arames (Wireframe):
  - Representa um objeto por arestas e pontos sobre o objeto;
  - Gera modelos ambíguos.
- Modelagem por superfície (década de 60):
  - Representa um objeto por meio de sua descrição matemática ;
  - Paramétrica  $\times$  Implícita.
- Modelagem por sólidos(década de 70):
  - Contém informações sobre o fechamento e conectividade do objeto.
- Modelagem de dimensão mista ou **Non Manifold**:
  - Permite modelar objetos com estruturas internas ou elementos pendentes de dimensão diferente.
  - Delimita o sólido por superfícies que não são necessariamente linear por partes

# Modelagem : Sólidos

Propriedade Requeridas.

## 1 Rigidez:

- Distância e ângulos fixos no espaço Euclidiano;
- A forma deve ser invariante sobre transformações rígidas.

## 2 Finitude:

- O objeto deve estar contido em uma porção limitada do espaço;
- O objeto deve ser descrito através de um número finito de símbolos;

## 3 Homogeneidade:

- O objeto não deve ter partes isoladas ou penduradas em sua fronteira.

## 4 Determinismo de fronteira:

- Deve ser possível descrever a fronteira e, conseqüentemente, o interior e exterior do objeto.

## 6 Fechamento sobre operações:

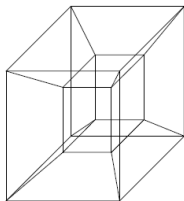
- O resultado de operações geométricas sobre objetos válidos devem ser ainda objetos válidos.

Quatro Categorias de Representação.

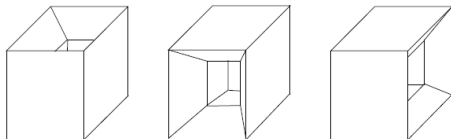
- 1 **Modelos de Arame:** (Wireframe);
- 2 **Modelos de Decomposição:** (BSP-trees, octrees, etc..);
- 3 **Modelos de Superfícies:** (Surface Modeling);
- 4 **Modelos de Sólido:** (Solid Modeling);
  - Modelos Construtivos (CSG - Constructive Solid Geometry);
  - Modelos de Fronteira (B-rep: Boundary Representation);
  - Modelos Híbridos (CSG e B-rep);
  - Modelos Baseados em Features (Feature Based Modeling);
  - Modelos de Dimensão Mista (Non Manifold).

# Modelagem : WireFrame

Ambiguidade de representação.



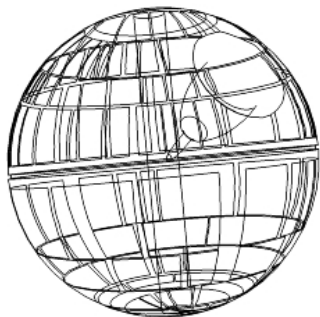
Modelo



Objetos representados

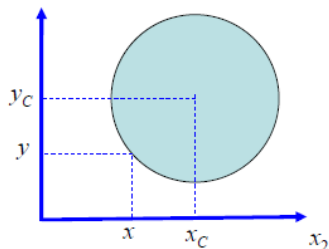
# Modelagem : WireFrame

## Primórdios da Computação Gráfica



# Modelagem:

## Representação Implícita



## Equação da circunferência

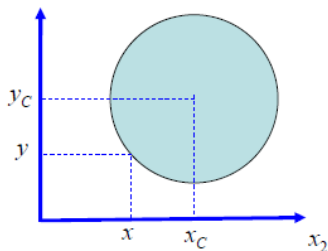
Centro  $(x_c, y_c)$  e raio  $r$ .

$$(x - x_c)^2 + (y - y_c)^2 = r^2$$



# Modelagem:

## Representação Paramétrica



Equação da circunferência

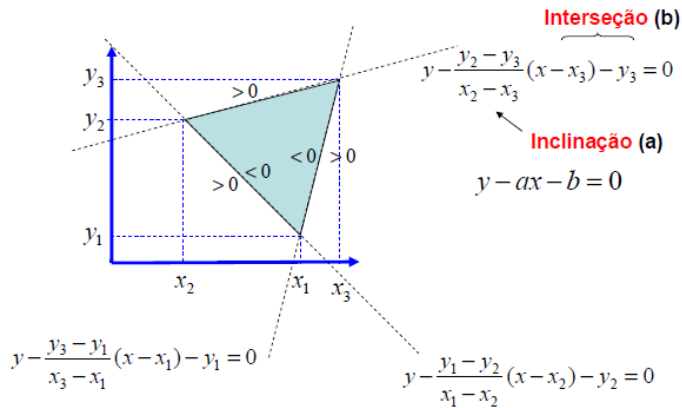
Centro  $(x_c, y_c)$  e raio  $r$ .

$$x = x_c + r \cos(\theta)$$

$$y = y_c + r \sin(\theta)$$

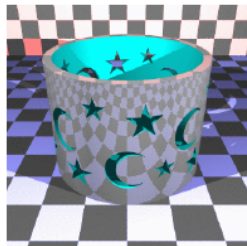
# Modelagem:

Representação Implícita de um Triângulo.



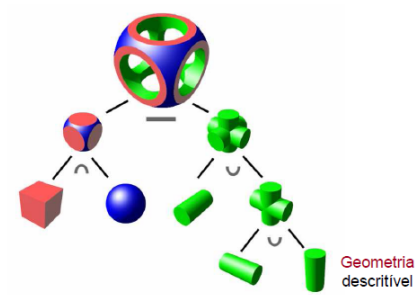
# Modelagem: Geometria Complexa

Não Há funções capazes de descrevê-las completamente

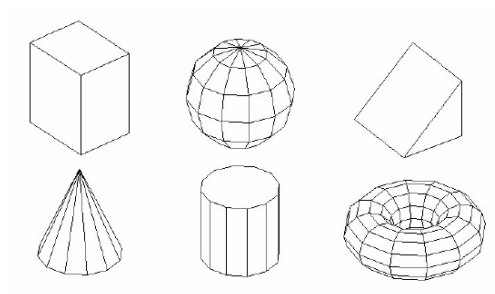


# Modelagem: Geometria Complexa

## Primitivas Geométricas + Operações Booleanas

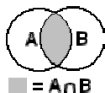
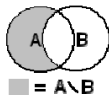
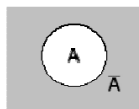
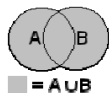


## Primitivas Geométricas + Operações Booleanas



# Modelagem: Geometria Complexa

## Primitivas Geométricas + Operações Booleanas



# Modelagem: Geometria Complexa

## Primitivas Geométricas + Operações Booleanas

Se  $f_1$  e  $f_2$  são funções implícitas que definem dois sólidos ( $f_1(p) \leq 0$  e  $f_2(p) \leq 0$ ). As operações de composição são definidas como:

$$\text{União: } f = f_1 \cup f_2 = \min \{f_1, f_2\}$$

$$\text{Intersecção: } f = f_1 \cap f_2 = \max \{f_1, f_2\}$$

$$\text{Diferença: } f = f_1 \setminus f_2 =_1 \cup \complement f_2 = \max \{f_1, -f_2\}$$

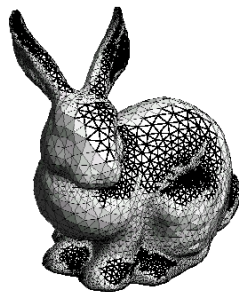
# Modelagem: Geometria Complexa

Duas Opções: Malhas, ou Decomposição Espacial



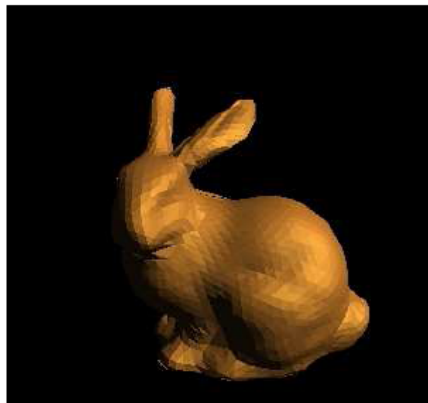
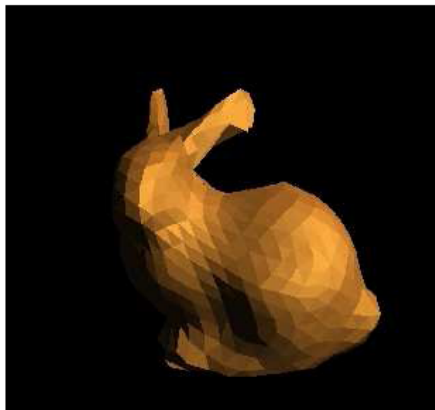


## Primeira Opção: Malhas



# Modelagem: Geometria Complexa

Primeira Opção: Malhas Poligonais



Área dos polígonos  $\rightarrow 0 \Rightarrow$  Forma  $\rightarrow$  Superfície Original

# Modelagem: Geometria Complexa

## Técnicas de Amostragem

- Manual (pelo digitalizador);
- Automática;

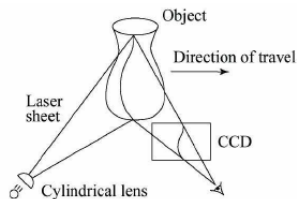
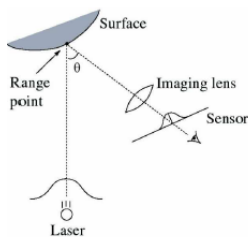


A Volkswagen Beetle becomes the subject of a 1970 simulation project. Ivan Sutherland(left) and assistants plot coordinates for digitizing the car.

# Modelagem: Geometria Complexa

## Técnicas de Amostragem

- Automática;



Problemas: concavidade e costura

# Modelagem: Geometria Complexa

## Técnicas de Amostragem

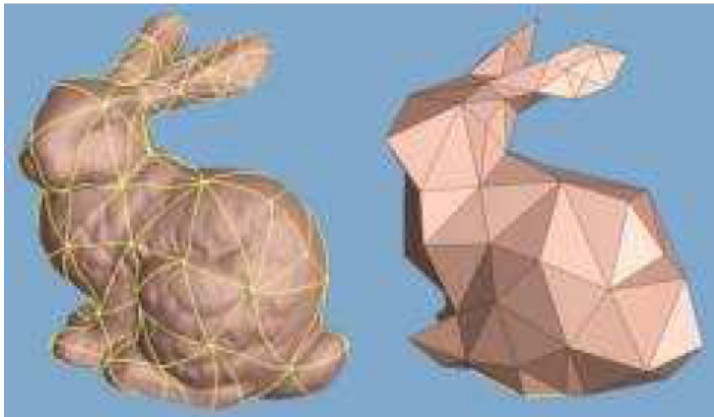
- Automática: SLAM (**S**imultaneous **L**ocalization **A**nd **M**apping);



Figura: Fonte: [Snavely,Seitz and Szeliski, 2006]

# Modelagem: Geometria Complexa

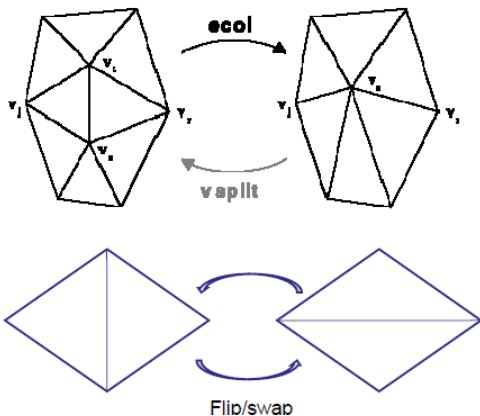
## Níveis de Resolução



# Modelagem: Geometria Complexa

## Níveis de Resolução

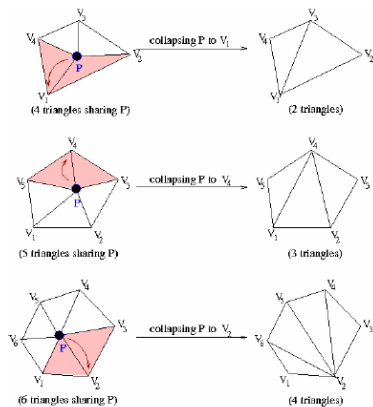
### Colapso de Arestas e Flip



# Modelagem: Geometria Complexa

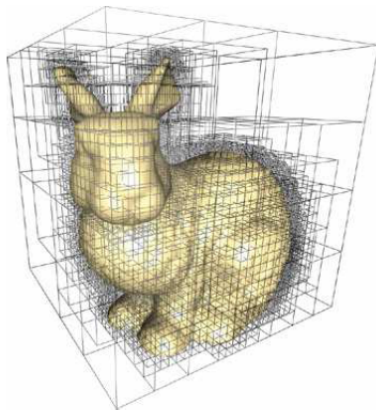
## Níveis de Resolução

### Colapso de Vértices





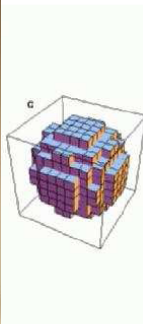
## Segunda Opção: Decomposição Espacial



# Modelagem: Geometria Complexa

## Segunda Opção: Decomposição Espacial

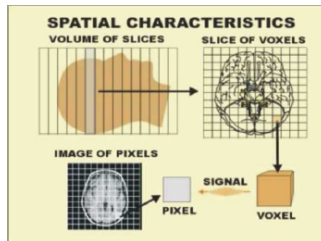
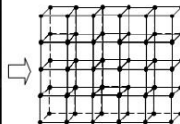
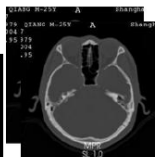
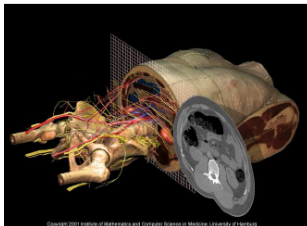
Realismo é mais difícil de ser alcançado



# Modelagem: Geometria Complexa

## Segunda Opção: Decomposição Espacial

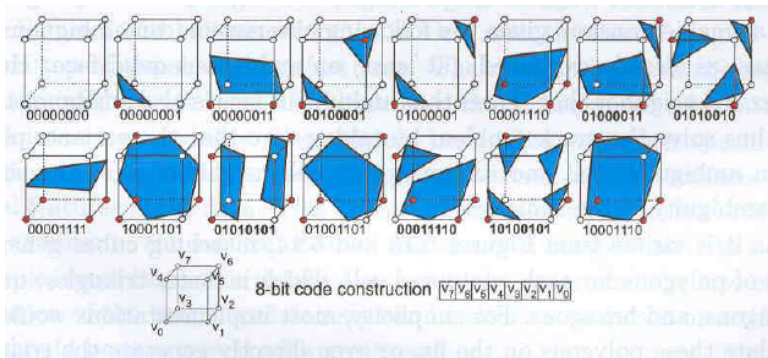
### Imagens Médicas 3D.



# Modelagem: Geometria Complexa

## Segunda Opção: Decomposição Espacial

Marching Cubes;

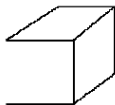
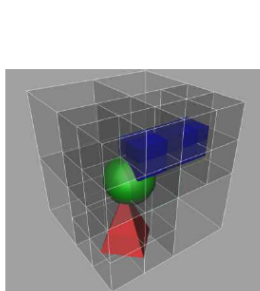


8 vértices  $\rightarrow 2^8$  possibilidades  $\rightarrow 15$  casos

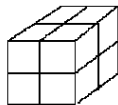
# Modelagem: Geometria Complexa

## Segunda Opção: Decomposição Espacial

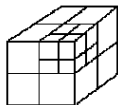
Representação por Octree;



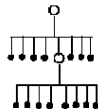
(root)



(1 level)



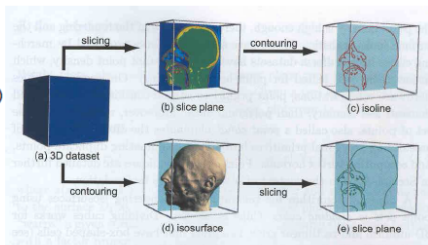
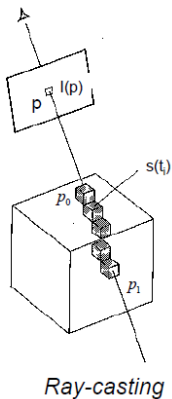
(2 levels)



# Modelagem: Geometria Complexa

## Segunda Opção: Decomposição Espacial

Processamento;



Redução em malhas poligonais

## Colocando a mão na Massa

- O POV-Ray, é um programa de ray tracing disponível para variadas plataformas de computação. Foi originalmente baseado no DKBTrace, escrito por David Kirk Buck e Aaron A. Collins. O POV-Ray é livre, com seu código-fonte disponível.
- Cenas em POV-Ray são descritas em uma linguagem de definição de dados denominada scene description language.
- Seu funcionamento, como OpenGL e semelhante ao de uma biblioteca C, uma vez que fornece uma série de funcionalidades.
- <http://www.povray.org/download/>

## Exercício

- 1 Modele um avião utilizando técnicas de CSG descritas nesta aula. Utiliza as primitivas geométricas disponíveis no POV-RAY;
- 2 Demonstre que a base  $\{C, n, v, u\}$  do espaço de câmera é ortonormal;



FIMM