

# Introdução às principais técnicas para a síntese de imagens por computador, a partir de modelos geométricos bidimensionais/tridimensionais

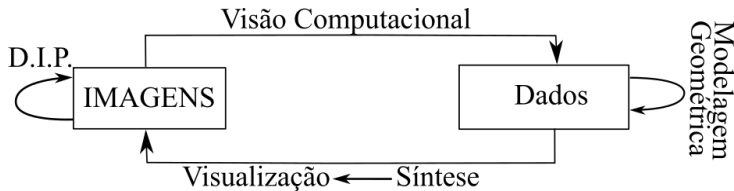
Prof. Marcos Amaris

Setembro - 2018



# Definição de síntese de imagens

- Síntese de imagens: é o processo de produção de imagens a partir de uma descrição tridimensional dos objetos nela contidos, tratamento de fontes de luz e posicionamento do observador
- Síntese de imagens é frequentemente confundida com a própria Computação Gráfica



- A Geometria (em grego antigo: geo- "terra", -metria "medida") é um ramo da matemática preocupado com questões de forma, tamanho e posição relativa desses objetos no espaços
- Ela estuda as formas presentes na natureza e as propriedades que essas formas possuem



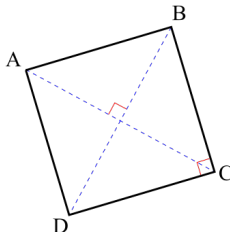
# Modelagem geométrica de objetos

- A síntese de imagens parte da descrição de figuras geométricas, produzindo uma imagem que pode, em última instância, ser visualizada em algum dispositivo
- Representação visual de objetos bidimensionais ou tridimensionais descritos através de especificações abstratas



# Pontos e Vértices

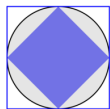
- O Ponto é um dos entes fundamentais da Geometria, junto com a reta e o plano, pois são considerados conceitos primários.
- Em geometria, um vértice é o ponto onde se encontram dois ou mais elementos unidimensionais (curvas, retas, entre outros)



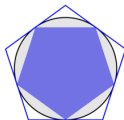
# Primitivas lineares bidimensionais

São as entidades básicas às quais as demais primitivas são reduzidas

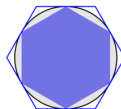
- Retas: elementos básicos dos modelos vetoriais;
- Poligonais: sequência de vértices unidos por retas chamadas arestas;
- Polígonos: áreas delimitadas por poligonais fechadas



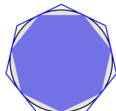
n=4



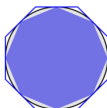
n=5



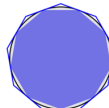
n=6



n=7



n=8



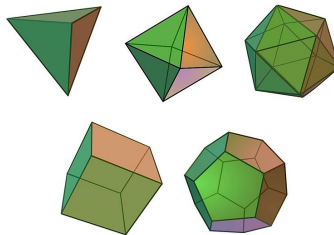
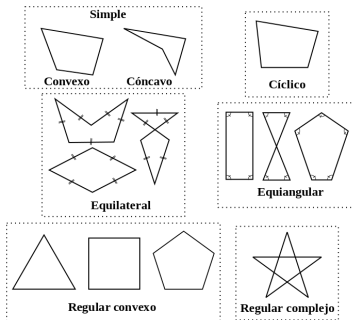
n=9



# Poliedros

Poliedros são sólidos geométricos limitados por polígonos. Um poliedro é dito regular quando obedece às três exigências seguintes:

- é convexo;
- é também poliedro de Platão;
- Os polígonos que o formam são regulares e congruentes.



# Figuras planas e espaciais

As figuras geométricas podem ser planas ou espaciais.

- O plano: Comprimento e largura, mas possui profundidade nula.
  - ▶ As figuras bidimensionais são denominadas de figuras planas.
- O espaço : Comprimento, largura e profundidade.
  - ▶ As figuras tridimensionais são denominadas espaciais e também chamadas de sólidos geométricos.





# Técnicas e modelos em CG

## Representação da fronteira

- Malha de polígonos
- Superfícies paramétricas
- Superfícies implícitas

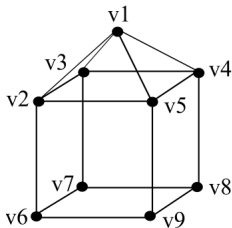
## Representação do Volume

- Voxels
- Quadrees e Octrees
- Constructive Solid Geometry (CSG)



# Malha de Polígonos

- Conjunto de vértices, arestas e faces planares (polígonos)
- Representação adequada para 'rendering' por placas gráficas: objetos gráficos padrão
- Forma padrão de representar objetos em CG



**Vértices (geometria)**

1	x1	y1	z1
2	x2	y2	z2
3	x3	y3	z3
4	x4	y4	z4
5	x5	y5	z5
6	x6	y6	z6
7	x7	y7	z7
8	x8	y8	z8
9	x9	y9	z9

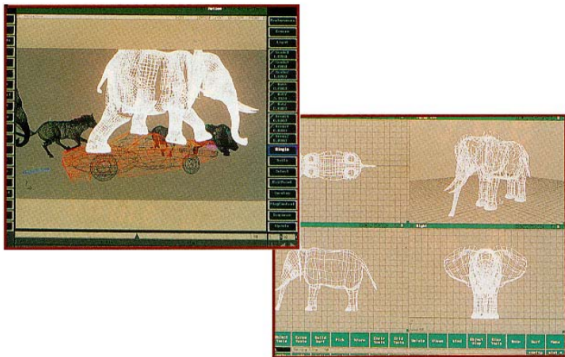
**Faces (topologia)**

1	v1	v4	v5	
2	v1	v5	v2	
3	v1	v2	v3	
4	v1	v3	v4	
5	v4	v3	v7	v8
6	v5	v4	v8	v9
7	v2	v5	v9	v6
8	v3	v2	v6	v7
9	v6	v9	v8	v7



# Malha de Polígonos

- Em superfícies curvas, a malha poligonal é uma aproximação para as superfícies. Ex. esfera, cone, cilindro
- Superfícies são decompostas em polígonos de modo a produzir uma representação poligonal aproximada



# Estrutura de Dados

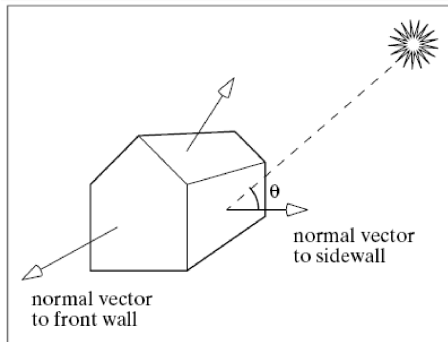
- Lista de faces, vértices e arestas
- Para cada vértice, armazena-se informação sobre sua posição
- Para cada aresta e face é descrita em termos dos seus vértices
- Para cada face, armazena-se a direção externa a ela
  - ▶ vetor normal à face poligonal
  - ▶ necessário para o rendering



# Vetores Normais

A direção normal à superfície determina seu brilho. É vantajoso associar um vetor normal à cada vértice.

- Facilita o processo de recorte, bem como o shading (sombras) de superfícies curvas



Fonte: Hill, Computer Graphics using OpenGL



# Propriedades das Malhas

- **Solidez:** o conjunto de suas faces delimita uma região finita
- **Conectividade:** pode-se percorrer uma sequência contínua de arestas entre quaisquer 2 vértices
- **Simplicidade:** representa um objeto sólido que não tem buracos
- **Planaridade:** as faces do objeto são polígonos planares
- **Convexidade:** qqr linha que conecta dois pontos está inteiramente contida no objeto



# Malhas Poligonais

- Rendering mais rápido
- Menor custo de armazenagem e transferência
- Manipulação mais simples
- Malhas de melhor qualidade para simulações numéricas...

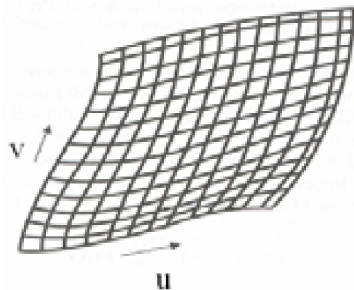


# Outras Representações de fronteira

## Superfícies paramétricas

São mais apropriadas para rerepresentar objetos tridimensionais com curvas

São também designadas por patches paramétricos bi-cúbicos e definem as coordenadas dos pontos de uma superfície curva através de 3 equações  $x = f(u, v)$ ,  $y = f(u, v)$  e  $z = f(u, v)$ .

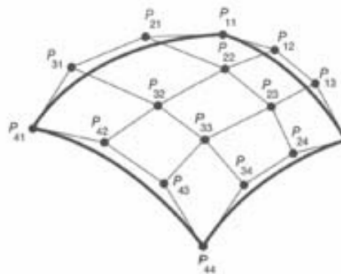




# Outras Representações de fronteira

## Superfícies paramétricas

- Cada equação possui dois parâmetros  $u$  e  $v$  e consiste num polinômio de 3º grau.
- Os limites da superfícies são definidos por curvas paramétricas cúbicas (por exemplo curvas Bézier).
- Em geral são usados 16 pontos de controle.
- Os pontos de controle definem a forma do patch.



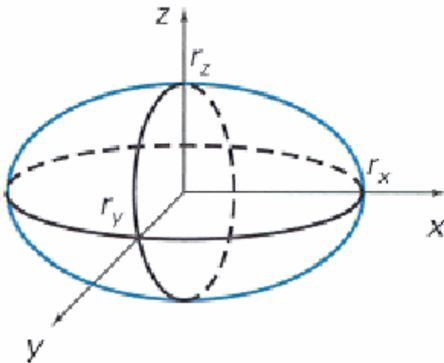
# Outras Representações de fronteira

## Superfícies Implícitas

São superfícies comuns, para as quais se conhece a função  $f(x, y, z)$  que as define.

É o caso das superfícies:

- Esfera
- Elipsoide
- Torus
- Paraboloide
- Hiperboloide
- etc



# Representação por Volume

## Voxels

### Voxels

O sólido é decomposto em células idênticas (voxels). A forma mais comum para os voxels é o cubo.

As propriedades do sólido são armazenadas com cada voxel:

- Volume
- Cor
- Densidade
- Temperatura
- etc

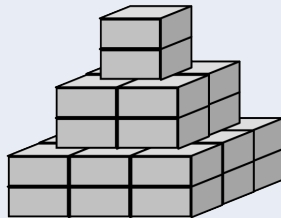
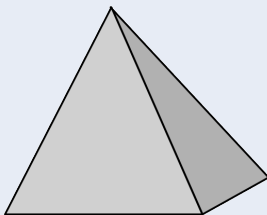


Figura original e a sua aproximação por Voxels



# Representação por Volume

## Quadtrees e Octrees

### Quadtrees e Octrees

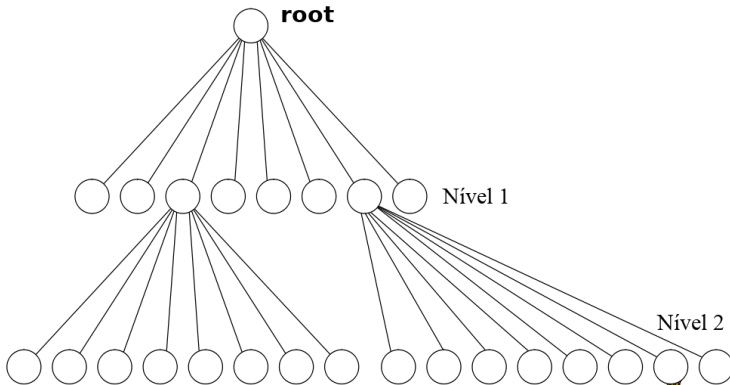
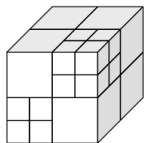
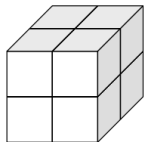
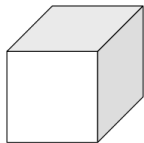
Quadtree e octree são árvore onde cada nó tem 4 e 8 filhos, respectivamente. O sólido é contido num cubo que depois é dividido em cubos menores

- 1 Cheio, caso o objeto ocupe todo o cubo.
- 2 Vazio, caso o objeto não ocupe nenhuma parte do cubo.
- 3 Cheio-vazio, caso o objeto ocupe parte do cubo.



# Representação por Volume

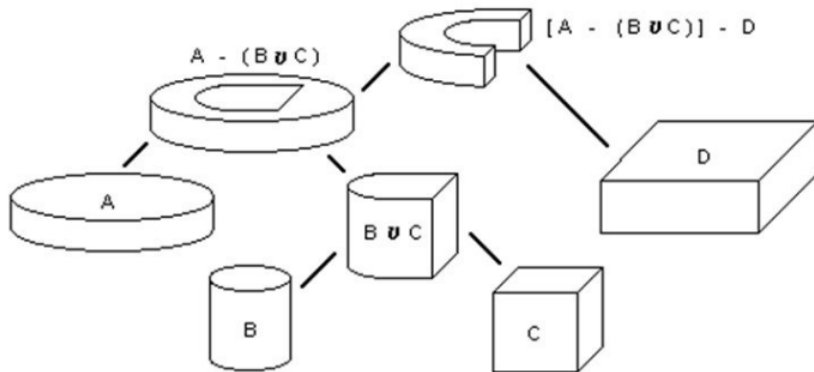
## Quadtrees e Octrees



# Representação por Volume

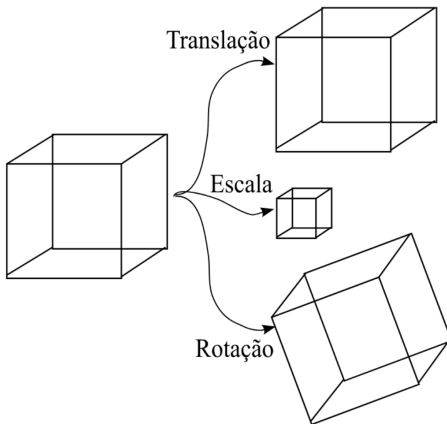
## Geometria Sólida Construtiva - CSG

O princípio básico implica a construção de um modelo complexo a partir de um conjunto de primitivas recorrendo-se de uma série de operações booleanas sequenciais alternadas ou não, tipicamente a União, Intersecção e a Diferença



# Transformações Geométricas

- Manipular e alterar interativamente as características dos objetos que compõem uma cena
- Os três tipos fundamentais de transformações geométricas são translação, rotação e escala



# Aplicações da Modelagem 3D

- Medicina
- Educação & Treinamento
- Desenho assistido por computador (CAD)
- Visualização científica
- Arte computadorizada
- Entretenimento (animação, filmes, jogos, ...)
- Processamento de Imagens
- Interfaces gráficas de usuário





# Ferramentas de Modelagem

Existem muitos programas de modelagem 3D

## Ferramentas de modelagem 3D

### Industriais

- Blender
- Maya
- D Studio Max.

### Iniciantes

- TinkerCAD
- Sketchup, mantido pela Google
- FreeCAD
- **OpenSCAD**

**OpenSCAD:** É um modelador baseado somente em scripts e usa sua própria linguagem de descrição;



# OpenSCAD

## CheatSheet

### Syntax

```
var = value;  
module name(...) { ... }  
name();  
function name(...) = ...  
name();  
include <...scad>  
use <...scad>
```

### 2D

```
circle(radius | d=diameter)  
square(size,center)  
square([width,height],center)  
polygon([points])  
polygon([points],[paths])  
text(text, size, font,  
    halign, valign, spacing,  
    direction, language, script)
```

### 3D

```
sphere(radius | d=diameter)  
cube(size, center)  
cube([width,depth,height], center)  
cylinder(h,r|d,center)  
cylinder(h,r1|d1,r2|d2,center)  
polyhedron(points, triangles, convexity)
```

### Transformations

```
translate([x,y,z])  
rotate([x,y,z])  
scale([x,y,z])  
resize([x,y,z],auto)  
mirror([x,y,z])  
multmatrix(m)  
color("colorname",alpha)  
color([r,g,b,a])  
offset(r|delta,chamfer)  
hull()  
minkowski()
```

### Boolean operations

```
union()  
difference()  
intersection()
```

### Modifier Characters

```
*      disable  
!      show only  
#      highlight / debug  
%      transparent / background
```

### Mathematical

```
abs  
sign  
sin  
cos  
tan  
acos  
asin  
atan  
atan2  
floor  
round  
ceil  
ln  
len  
let  
log  
pow  
sqrt  
exp  
rands  
min  
max
```



# OpenSCAD

## CheatSheet

### Functions

[concat](#)  
[lookup](#)  
[str](#)  
[chr](#)  
[search](#)  
[version](#)  
[version\\_num](#)  
[norm](#)  
[cross](#)  
[parent\\_module](#)(idx)

### Other

[echo](#)(...)  
[for](#) (i = [start:end]) { ... }  
[for](#) (i = [start:step:end]) { ... }  
[for](#) (i = [...],...) { ... }  
[intersection for](#)(i = [start:end]) { ... }  
[intersection for](#)(i = [start:step:end]) { ... }  
[intersection for](#)(i = [...],...) { ... }  
[if](#) (...) { ... }  
[assign](#) (...) { ... }  
[import](#)("...stl")  
[linear extrude](#)(height,center,convexity,twist,slices,scale)  
[rotate extrude](#)(angle,convexity)  
[surface](#)(file = "...dat",center,convexity)  
[projection](#)(cut)  
[render](#)(convexity)  
[children](#)([idx])

### List Comprehensions

[Generate](#) [ for (i = range|list) i ]  
[Conditions](#) [ for (i = ...) if (condition(i)) i ]  
[Assignments](#) [ for (i = ...) let (assignments) a ]

### Special variables

[\\$fa](#) minimum angle  
[\\$fs](#) minimum size  
[\\$fn](#) number of fragments  
[\\$t](#) animation step  
[\\$vpr](#) viewport rotation angles in degrees  
[\\$vpt](#) viewport translation  
[\\$vpd](#) viewport camera distance  
[\\$children](#) number of module children



- **Introdução à computação gráfica** Roberto Scalco. 2005. Capítulo 3. Edit. Maua, Escola de Engenharia
- **Foundations of 3D Computer Graphics.** Steven Jacob Gortler. Edit. MITPRESS



**Atividade:** Realizar o desenho de 3 objetos da natureza, usando OpenSCAD e suas diferentes formas geométricas e funções de transformação



- Ler o Artigo: Introdução à Computação Gráfica. Isabel Harb Manssour e Marcelo Cohen.

## Até a Próxima :)

