universidade federal de pelotas
CDTec
centro de desenvolvimento tecnológico

computação
Universidade Federal de Pelotas

Computer Graphics
Today: Graphic Pipeline

Prof. PhD Rafael P. Torchelsen
rafael.torchelsen@inf.ufpel.edu.br

Pipeline

computação
Universidade Federal de Pelotas

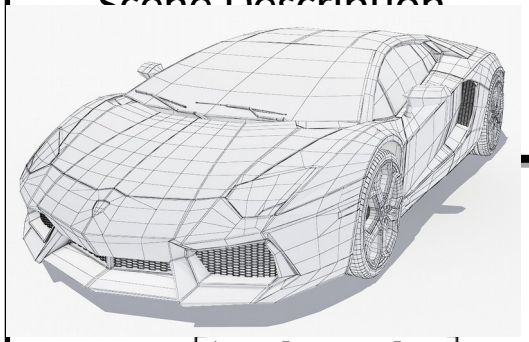
Graphic Pipeline

computação
Universidade Federal de Pelotas

You know this?

You know this!

Black Box



Input

Processing

Output/Input

Image

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

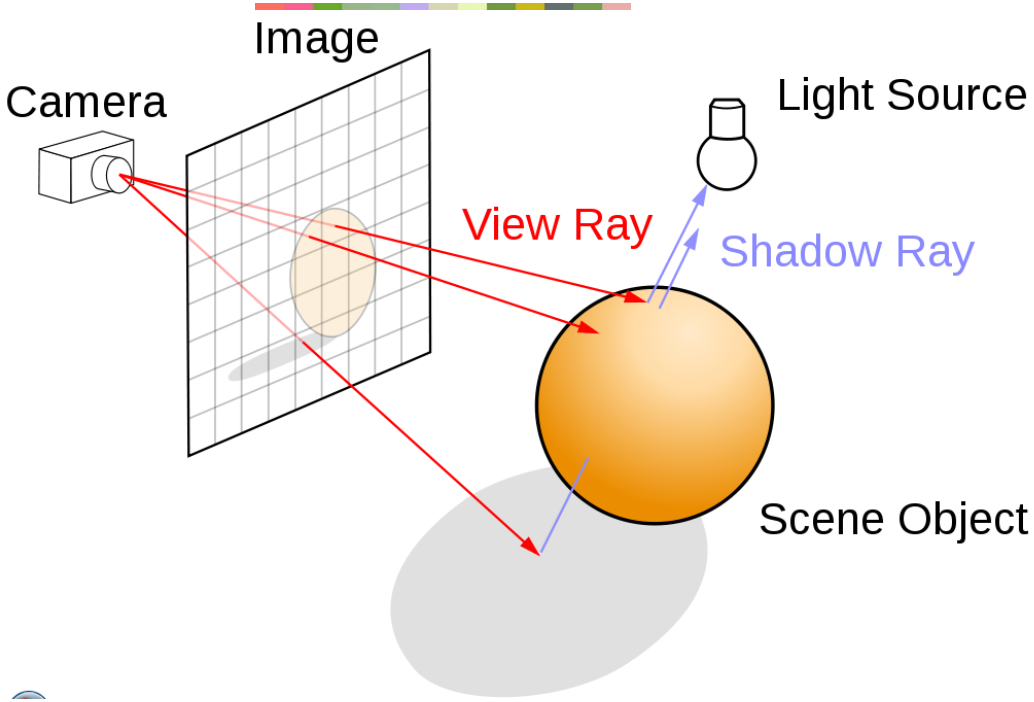
$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}$$

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

Vectors and Matrixes

Render Overview

computação
Universidade Federal de Pelotas



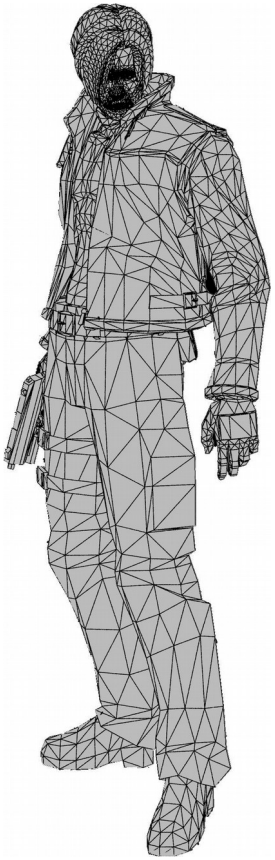
The color must come from somewhere

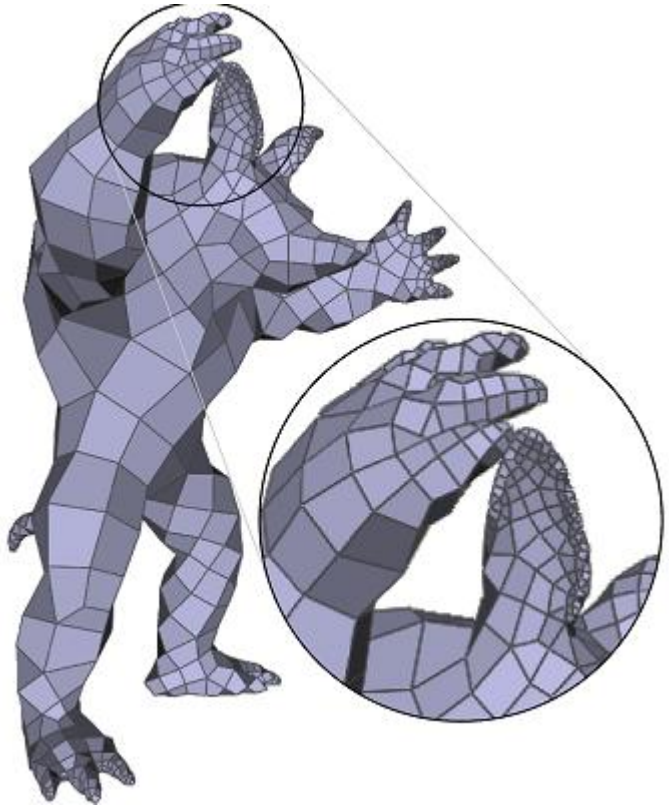
Scene (Objects, Lights, Camera)

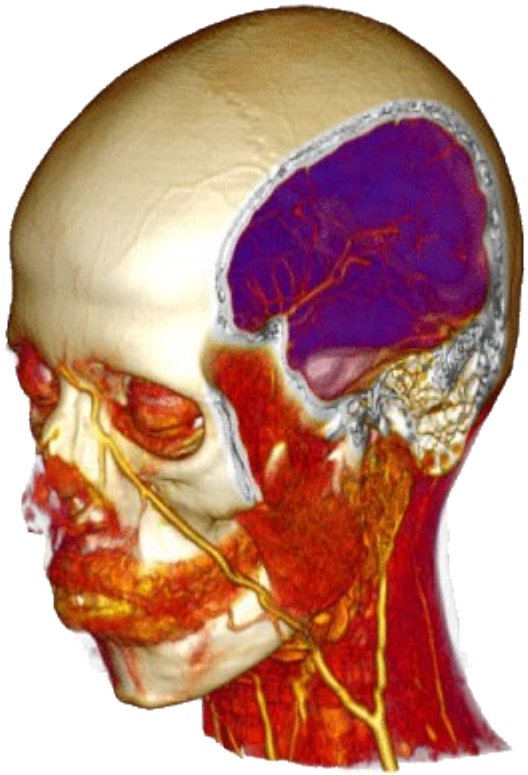
How we describe a scene?

Common ways to describe an object

computação
Universidade Federal de Pelotas







Triangle Mesh

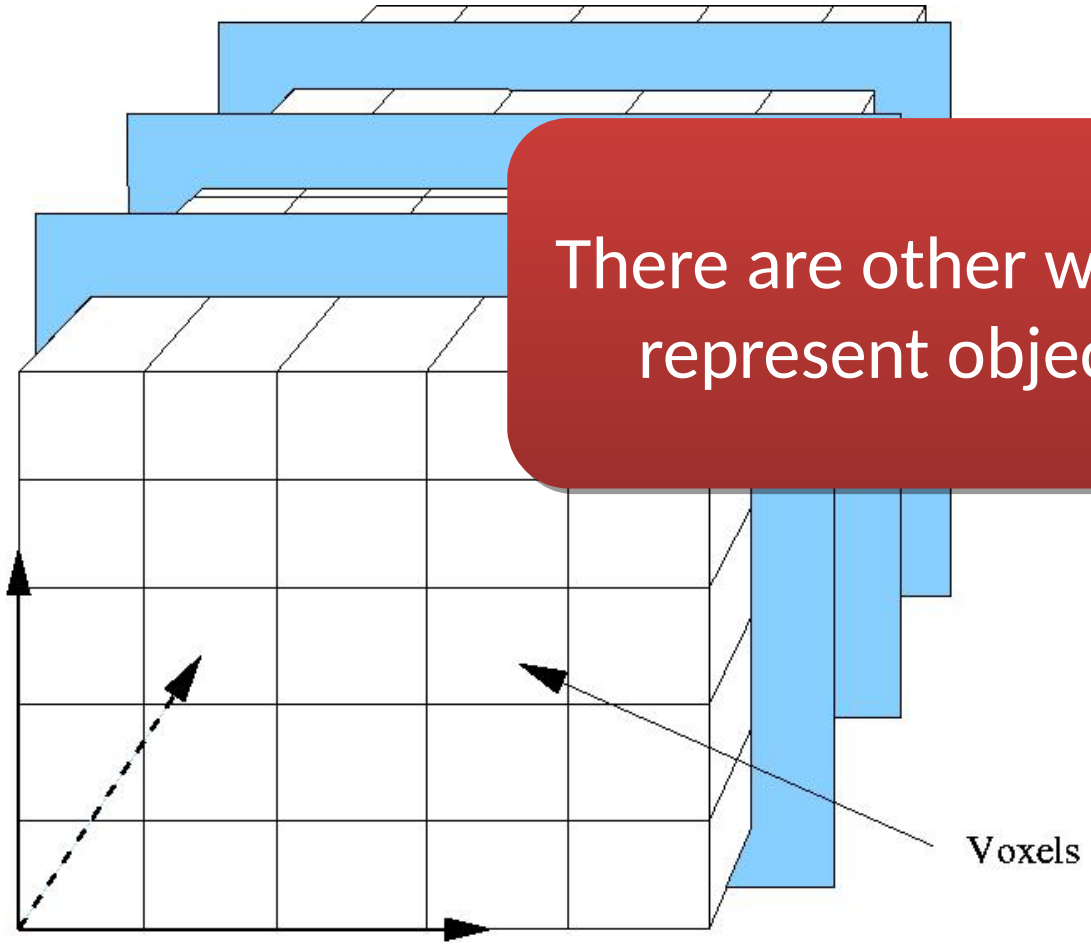
Quad Mesh

Volume

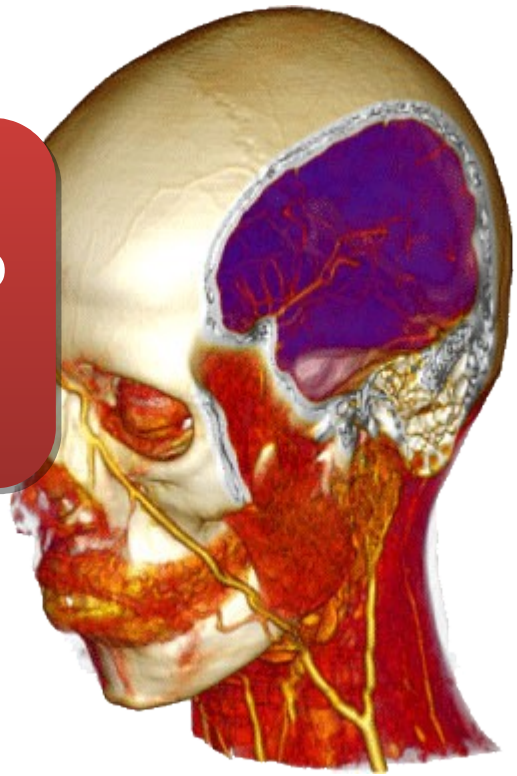
Surface only

Common ways to describe an object

computação
Universidade Federal de Pelotas



There are other ways to represent objects!

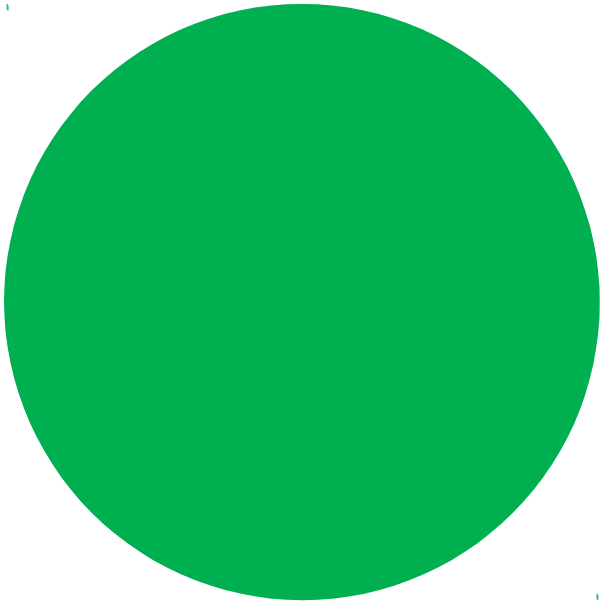


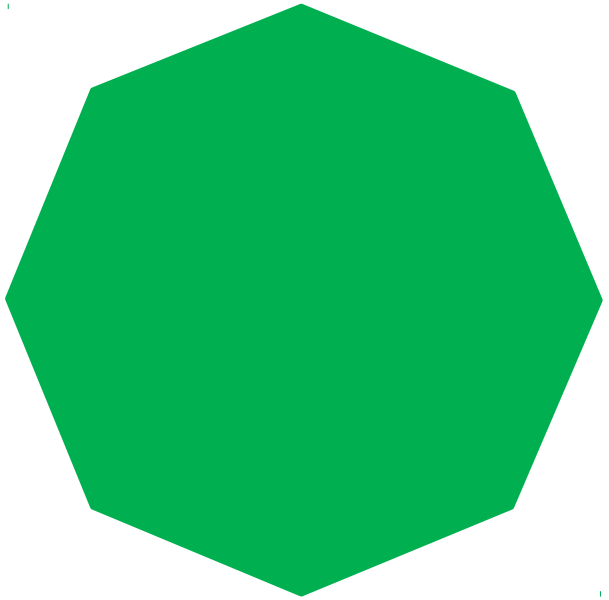
Volume

Triangle Mesh 2D

computação
Universidade Federal de Pelotas

- Discretization
- Connection



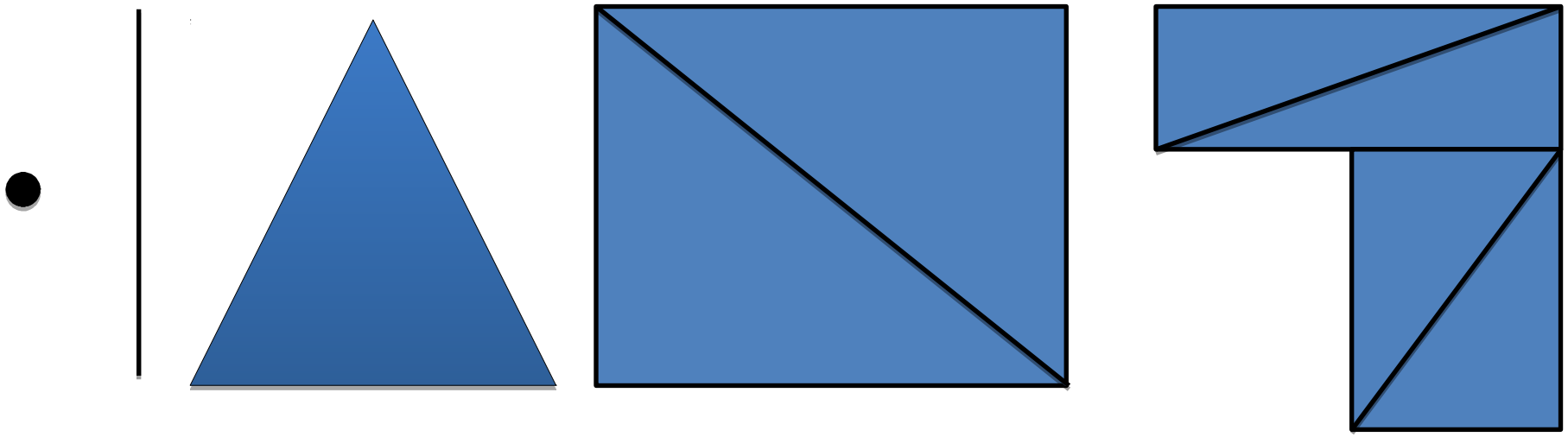


Approximation, but we can increase the resolution!

Why Triangles?

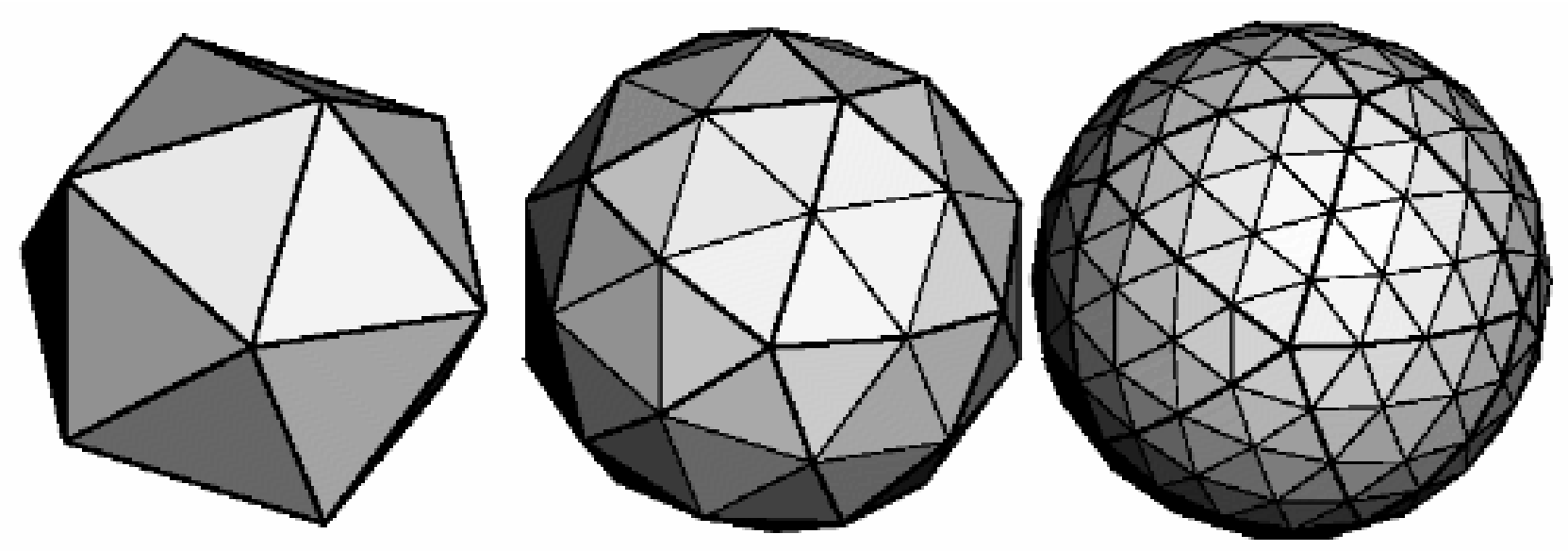
computação
Universidade Federal de Pelotas

- The simplest polygon that has area
- Any polygon can be represented by triangle**s**
- Always planar



Triangle Mesh 3D

computação
Universidade Federal de Pelotas



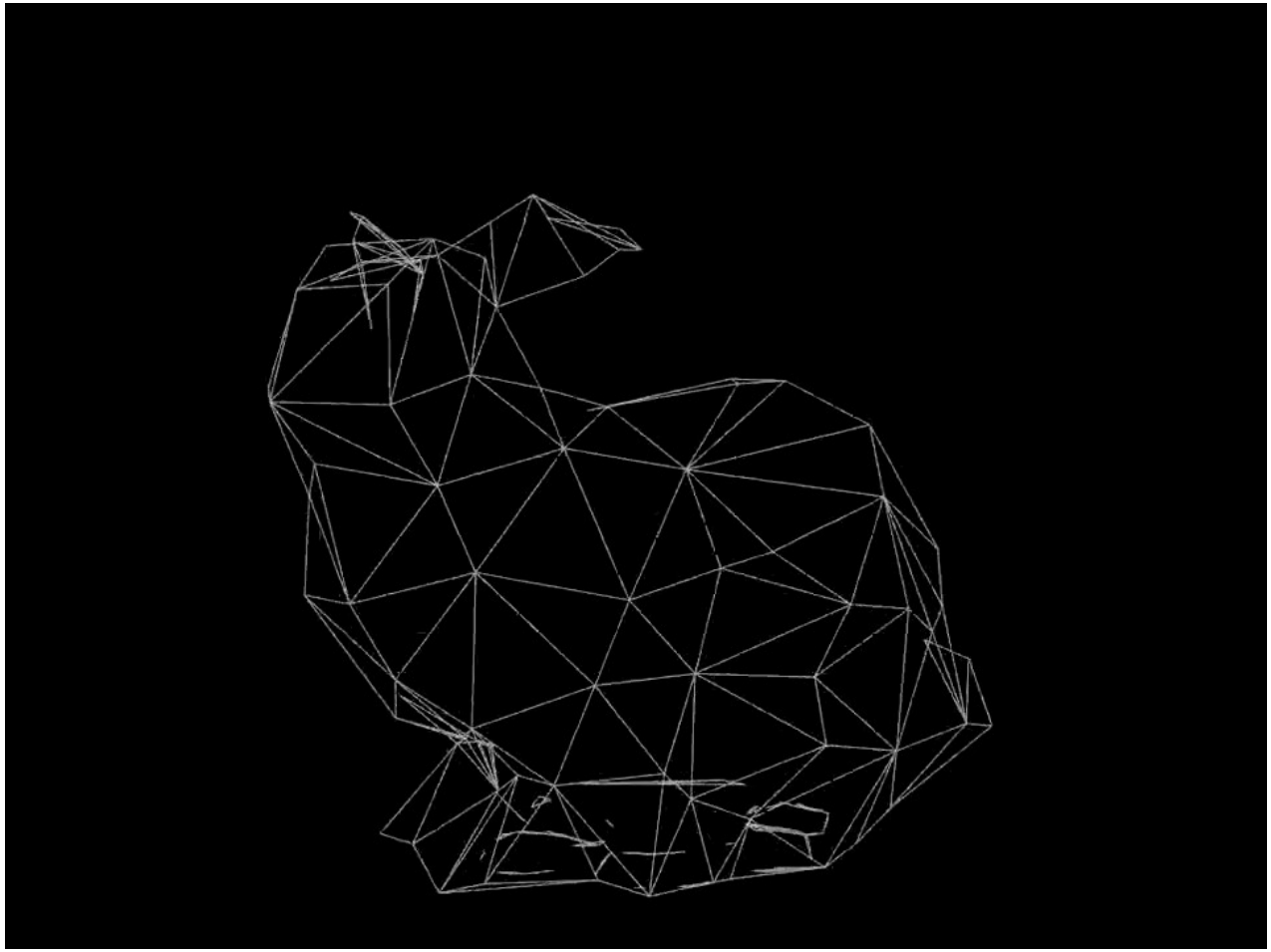
Empty shell!

Resolution

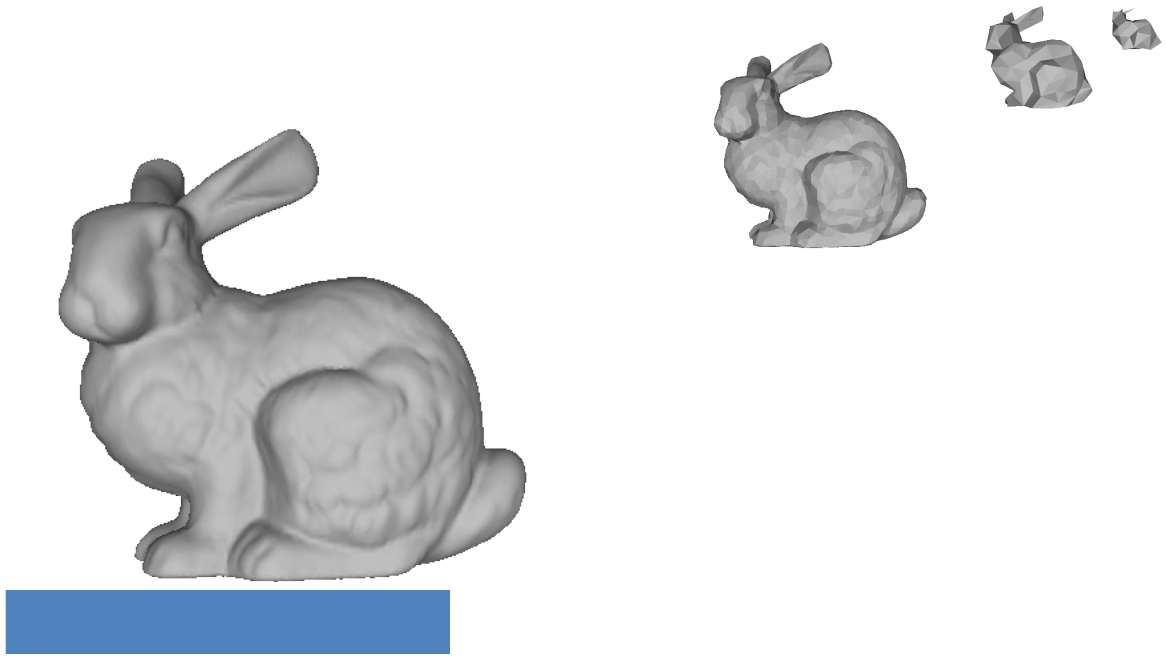
computação
Universidade Federal de Pelotas

Level of detail - LOD

computação
Universidade Federal de Pelotas

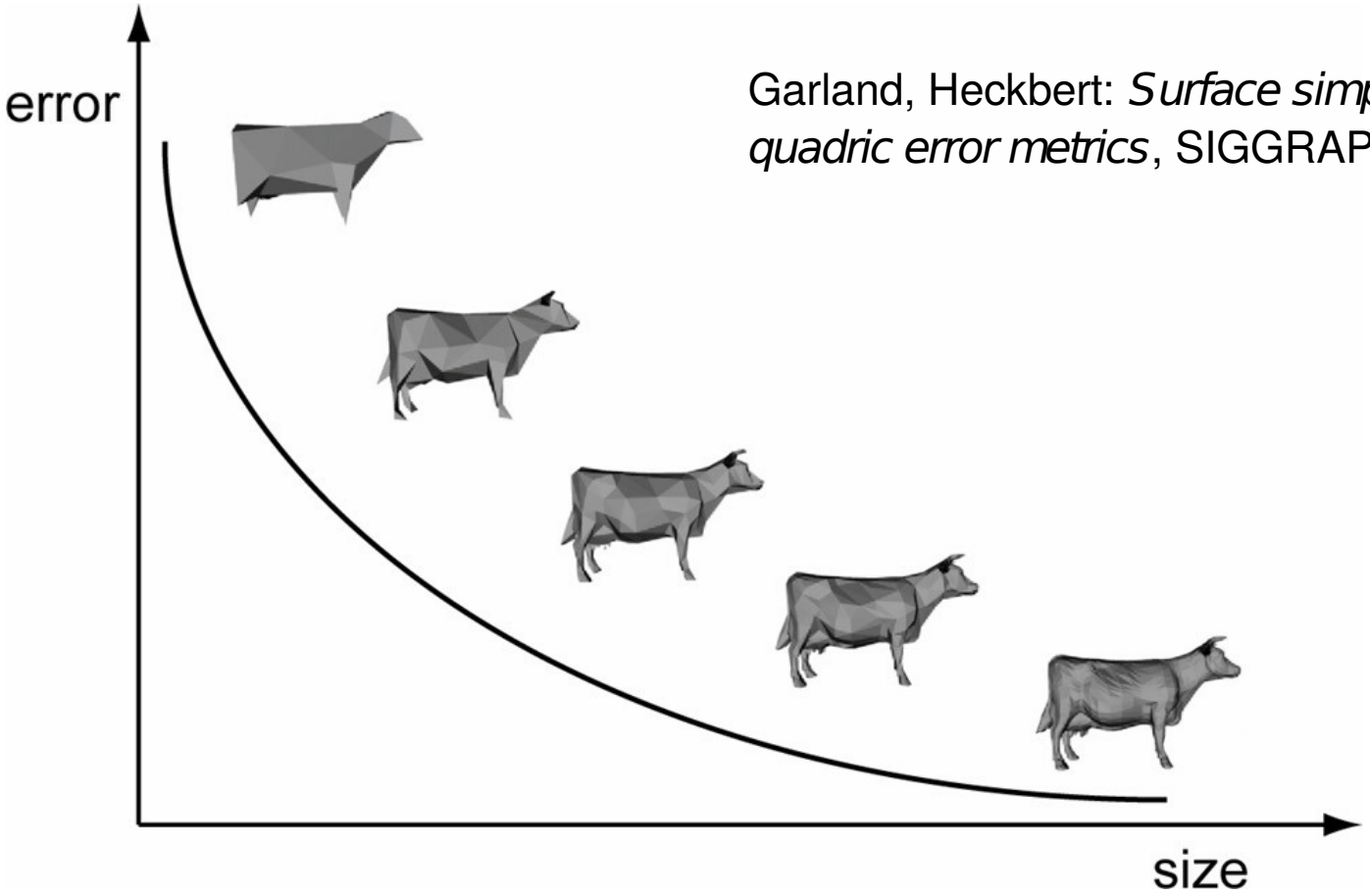


Distant objects use coarser LODs

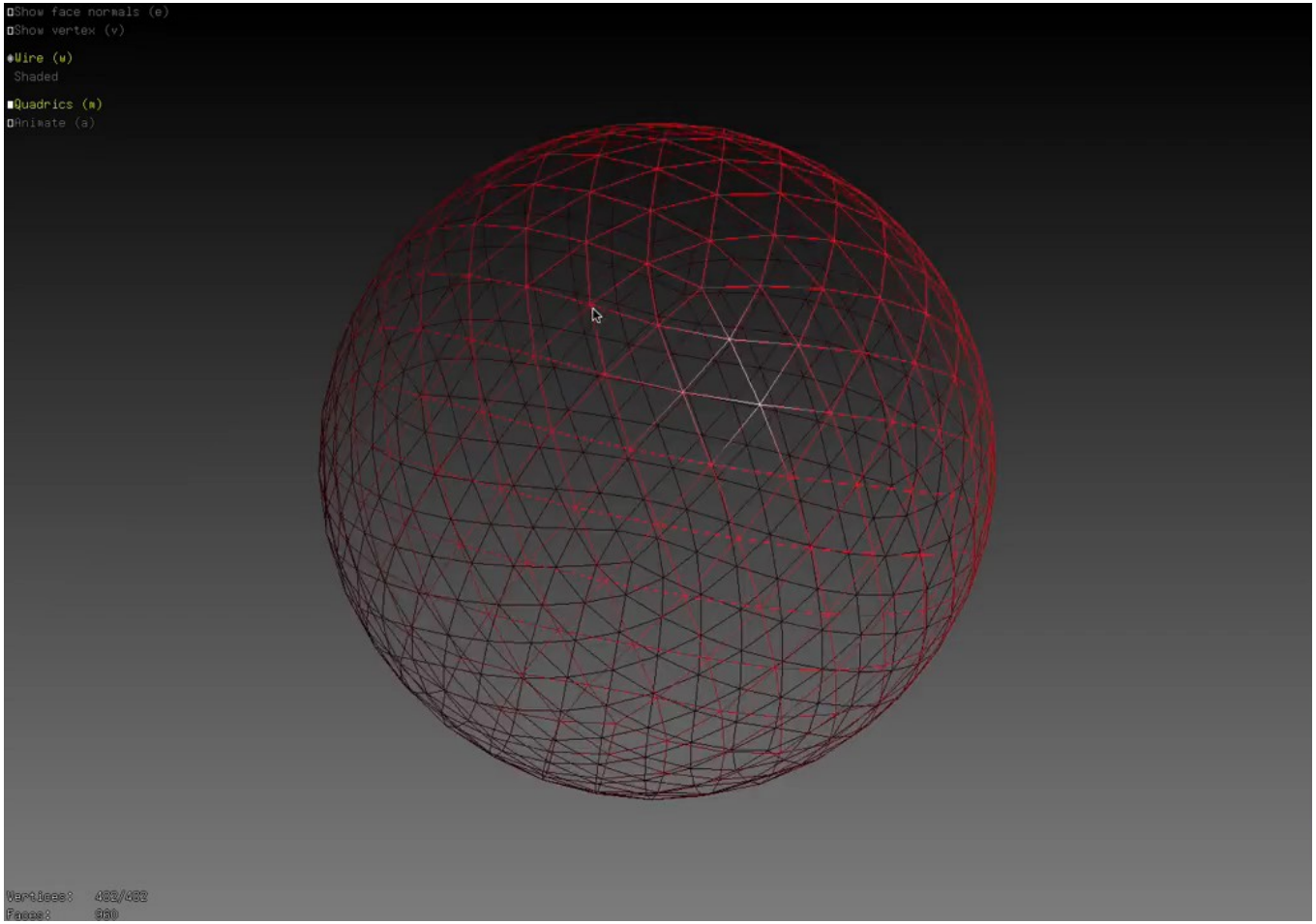


Error vs Size

computação
Universidade Federal de Pelotas

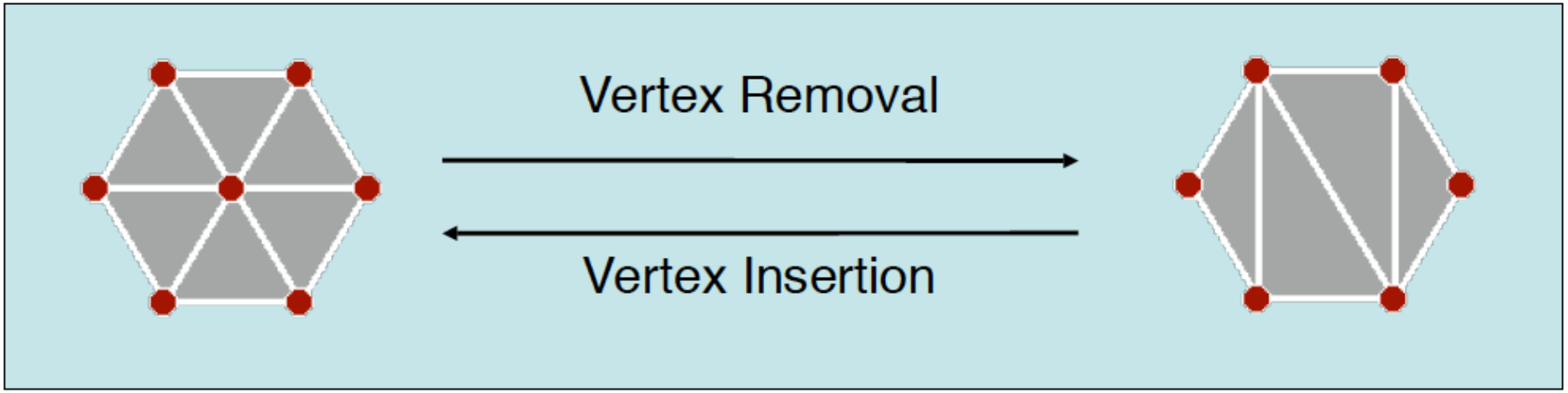


Garland, Heckbert: *Surface simplification using quadric error metrics*, SIGGRAPH 1997


Show face normals (e)
Show vertex (v)
Wire (w)
Shaded
Quadrics (m)
Animate (a)
Vertices: 482/482
Faces: 960

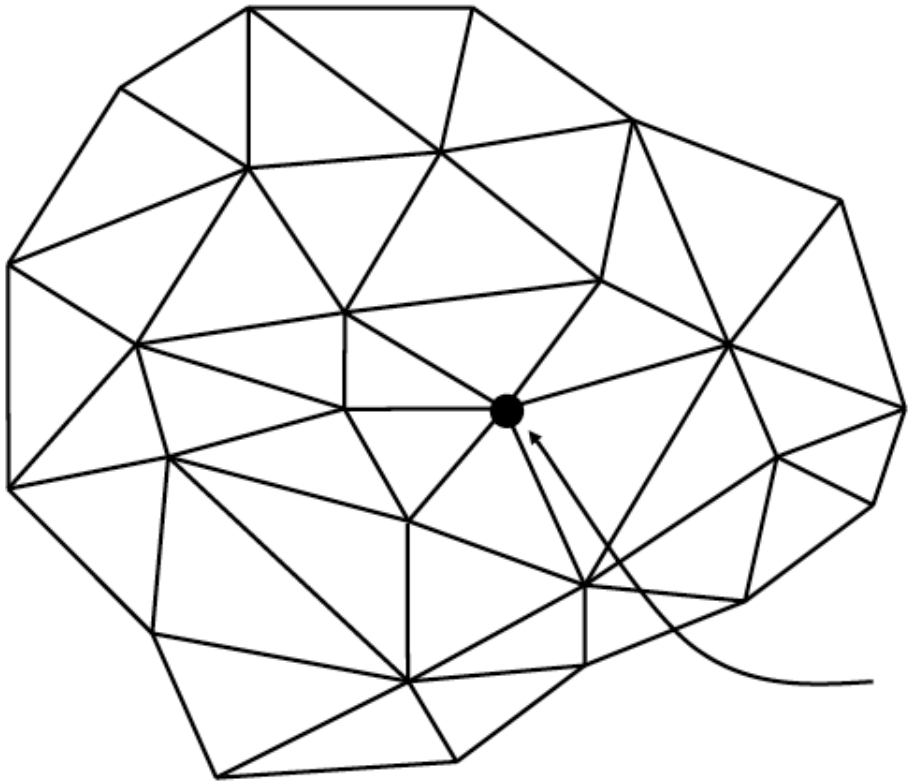
Vertex removal

computação
Universidade Federal de Pelotas

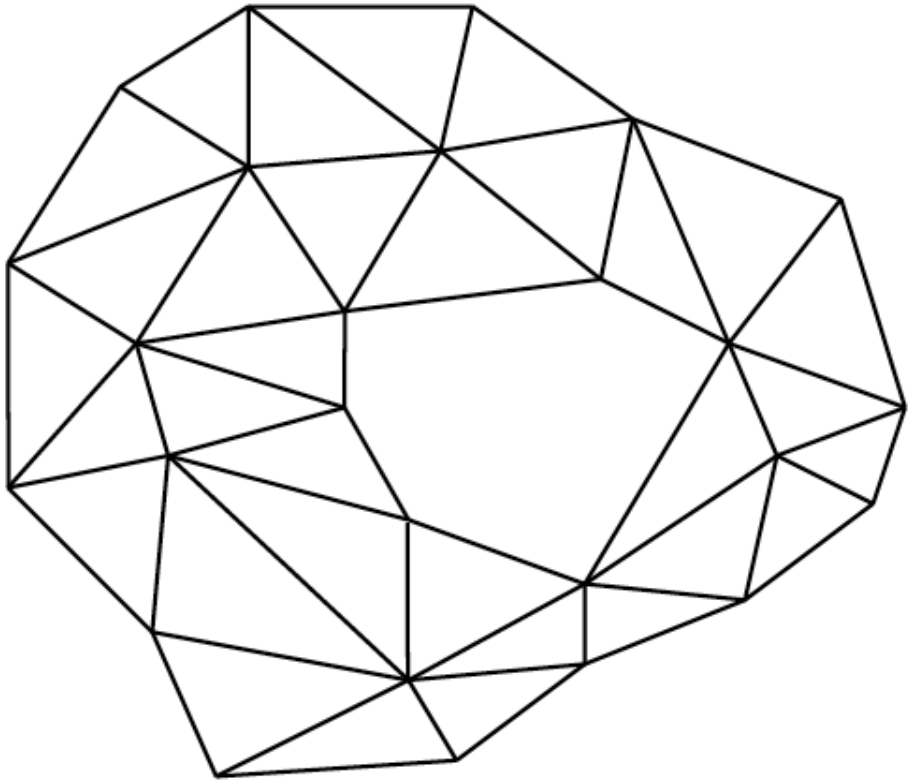


Algorithm overview

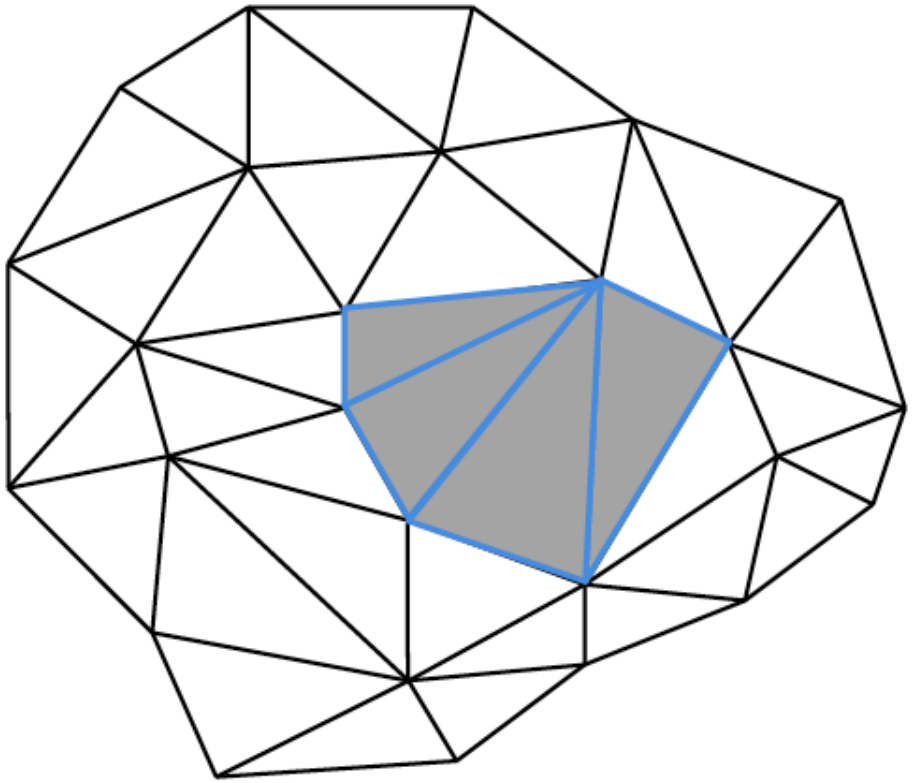
1 – Select a vertex to remove (Priority queue)
2 – Remove the vertex
3 – Re-triangulate the hole



Select an element
to be eliminated



Remove the selected triangles, creating the hole



Fill the hole
with triangles

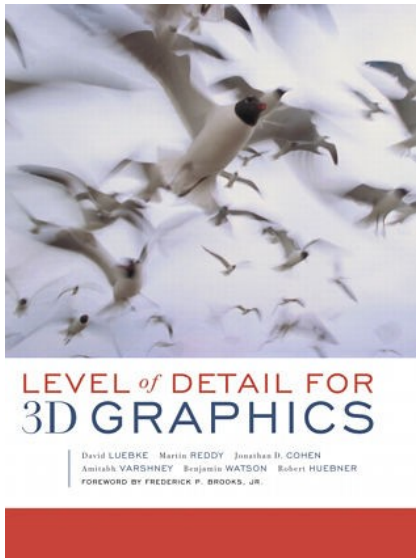
References

computação
Universidade Federal de Pelotas

- **Geometric Modeling Based on Polygonal Meshes**
  Mario Botsch, Mark Pauly, Leif Kobbelt, Pierre Alliez, Bruno Levy, Stephan Bischoff, Christian Rössl
  Eurographics 2008 Course Notes

Moodle: course notes and slides

Read: section 2 and 9

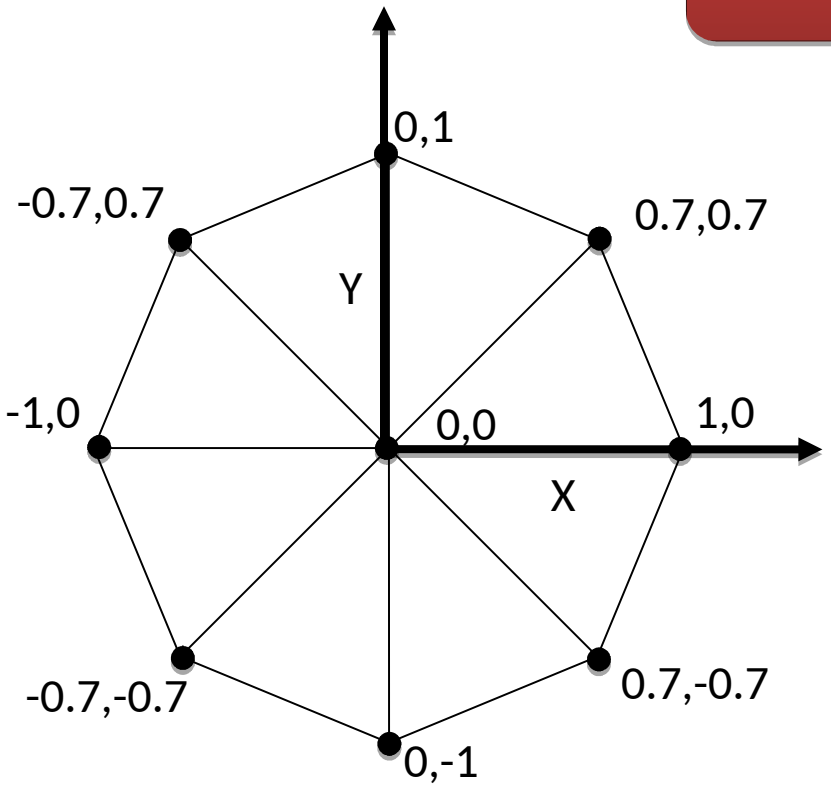


http://lodbook.com/
http://lodbook.com/course/2003/
http://lodbook.com/models/

The Origin

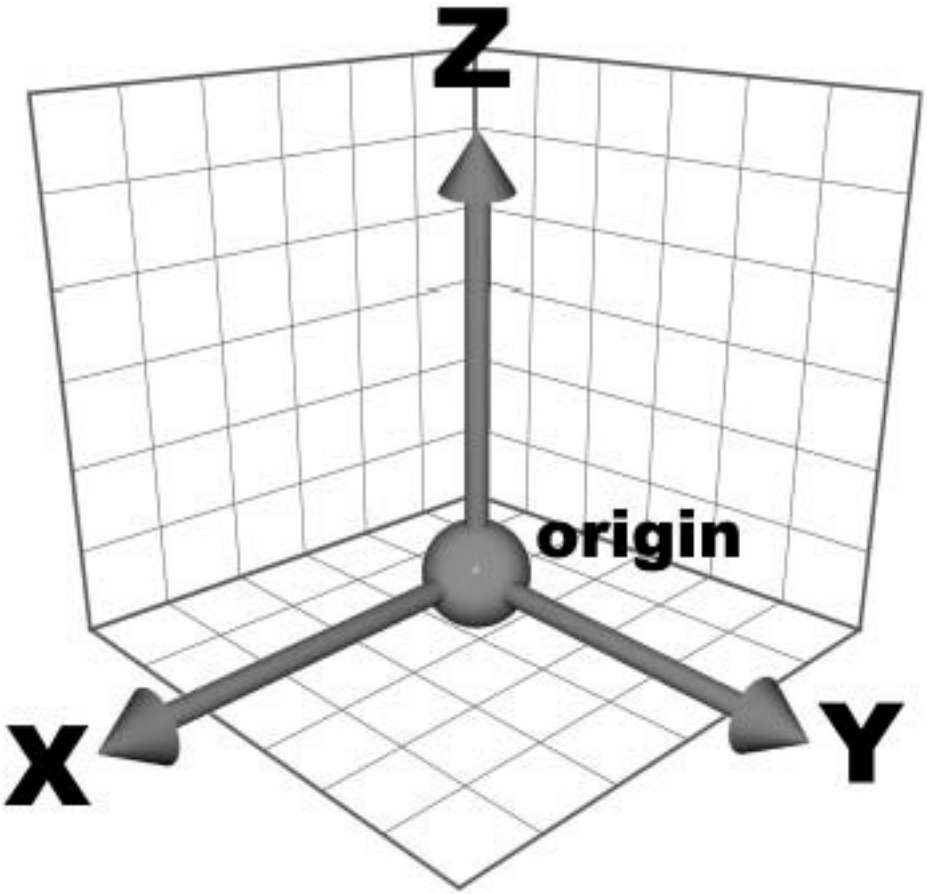
computação
Universidade Federal de Pelotas

The origin can be any point!



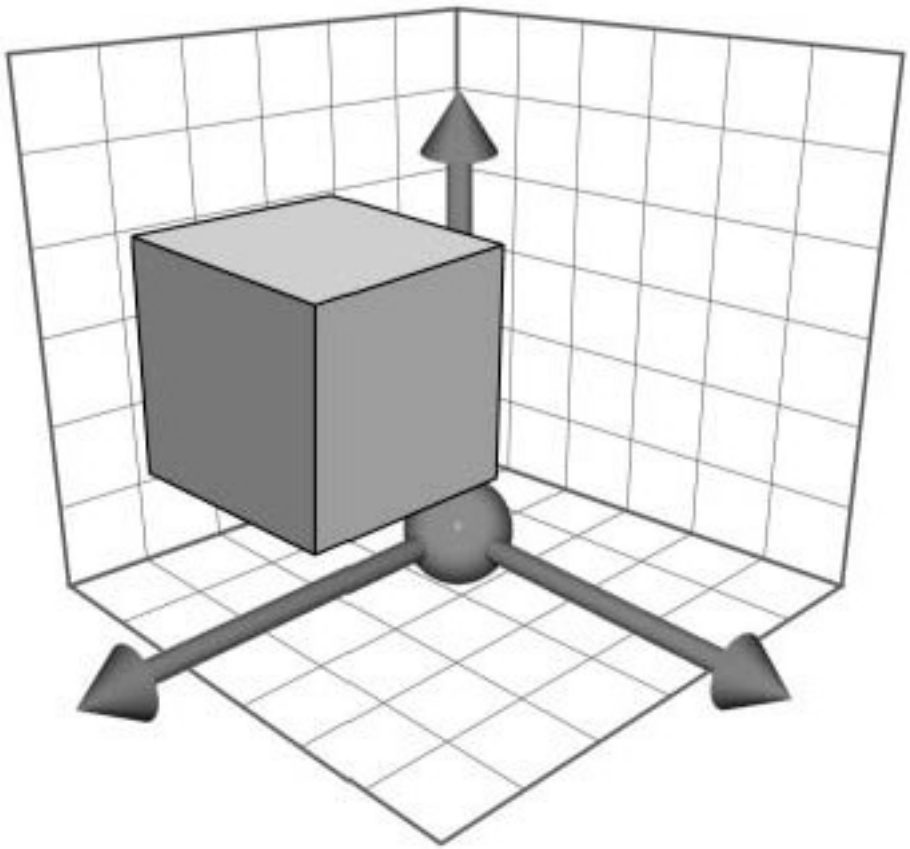
Virtual Space

computação
Universidade Federal de Pelotas

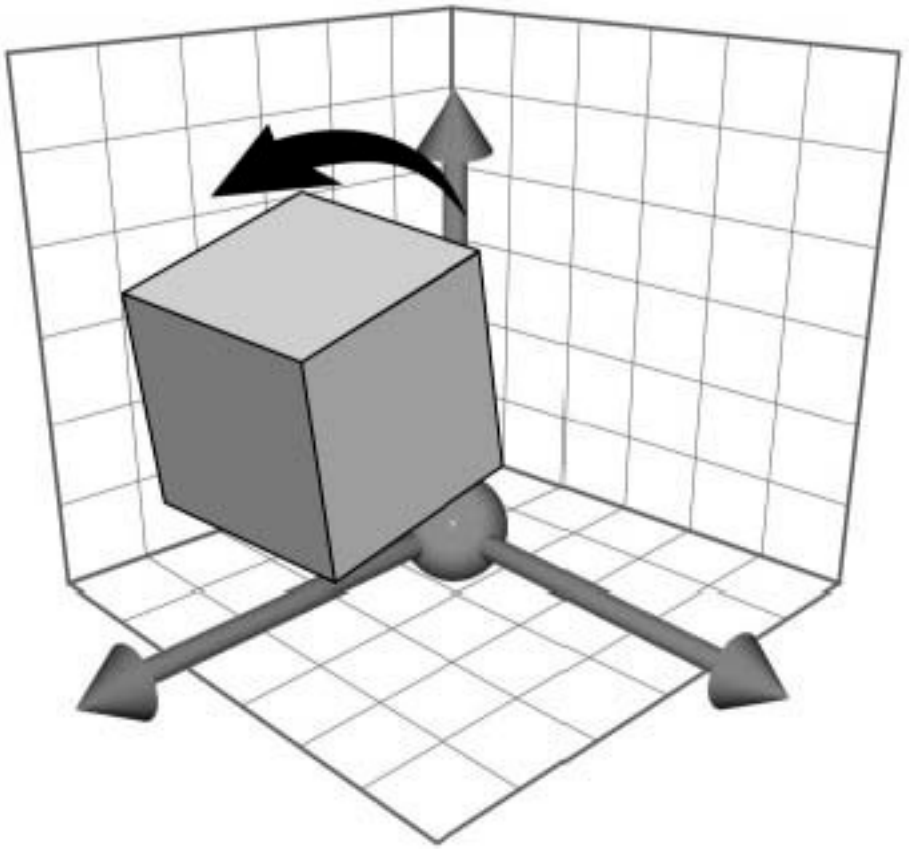


Transformations

computação
Universidade Federal de Pelotas



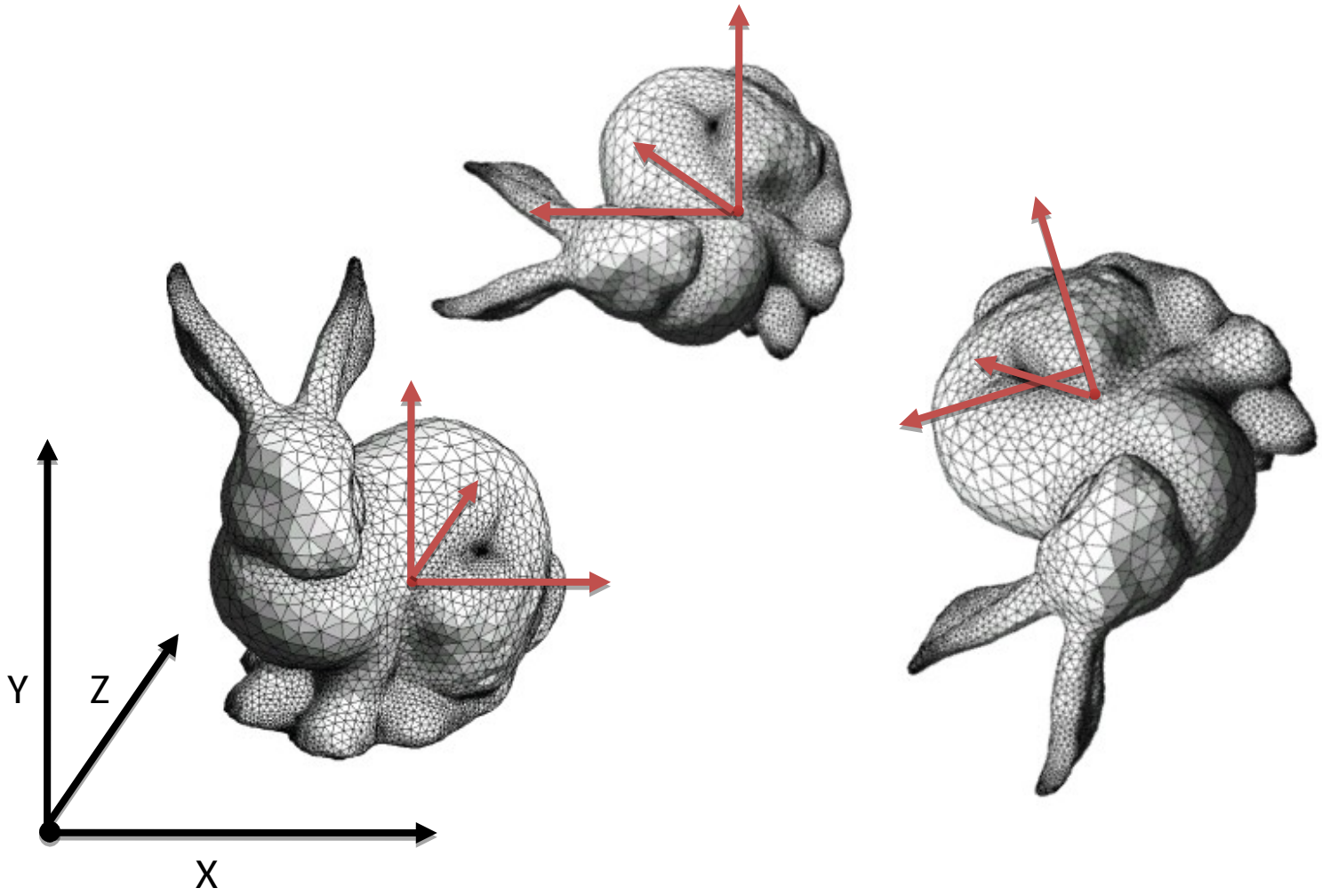
Translation



Rotation

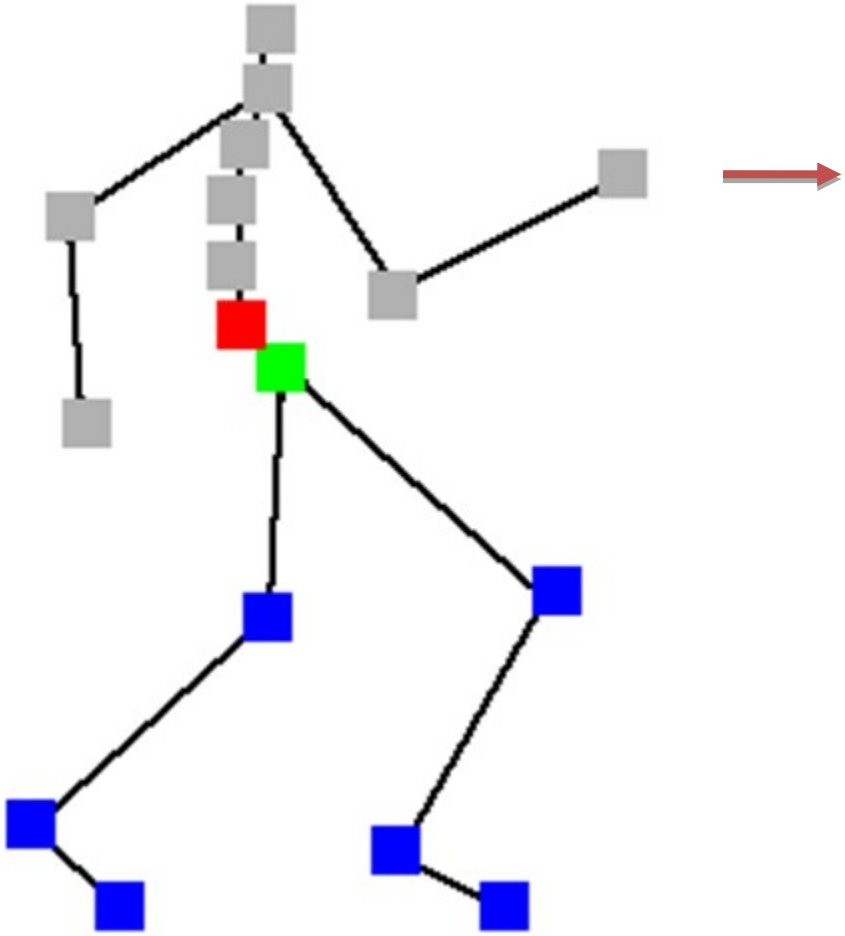
Hierarchical Transformations

computação
Universidade Federal de Pelotas



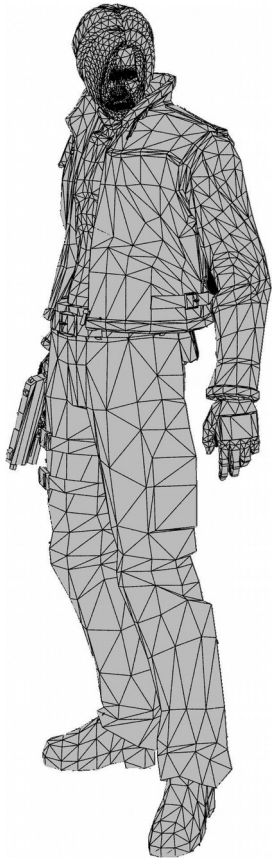
Hierarchical Transformations

computação
Universidade Federal de Pelotas



Surface vs. Light vs. Color

computação
Universidade Federal de Pelotas



Surface Description

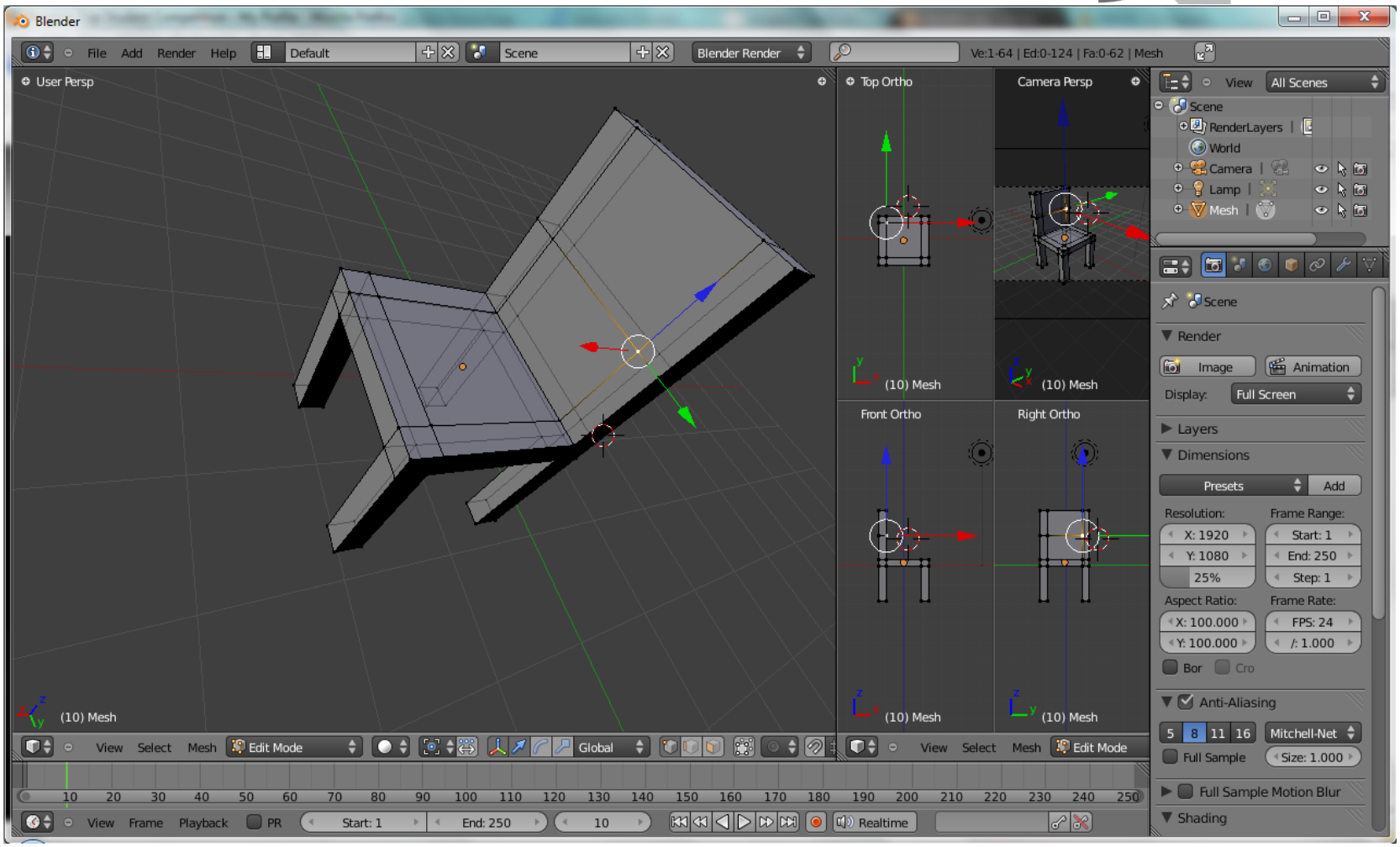
Light

Color

We will see this in a few weeks

Blender

computação
Universidade Federal de Pelotas



Home-Work: Find a tutorial about Blender (Youtube)

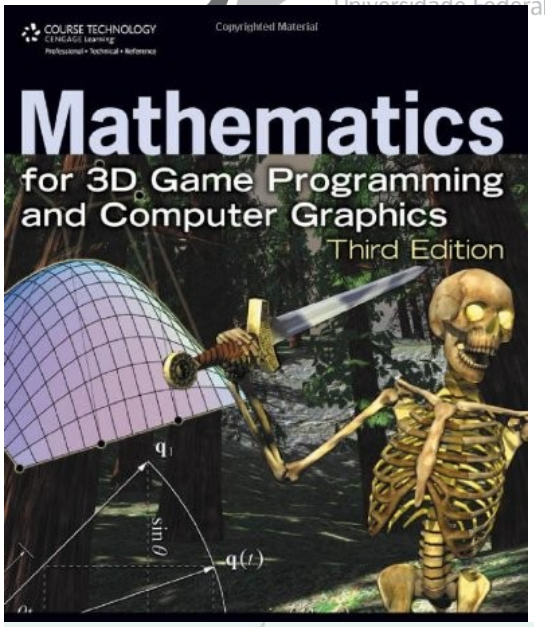
You need to know

- Planes
- Lines
- Points
- Vectors
- Matrixes

Álgebra Linear e Geometria Analítica

↓

Computação Gráfica



Overview

computação
Universidade Federal de Pelotas

http://www.gamedev.net/page/resources/_/technical/graphics-programming-and-theory/introduction-to-the-graphics-pipeline-r3344