



# Survey de técnicas para simulação de malhas detalhadas: Análise e Comparação

**Bruno Pereira Evangelista**  
[bpevangelista@gmail.com](mailto:bpevangelista@gmail.com)

**Orientador:**  
**Marcelo Souza Nery**  
[msnery@pucminas.br](mailto:msnery@pucminas.br)

**PUC Minas**

# Sumário

---

- **Introdução**
- **Motivação**
- **Objetivo e Metodologia**
- **Efeitos visuais desejados**
- **Processo renderização**
- **Análise das técnicas**
- **Comparação das técnicas**
- **Conclusão**

# Introdução

---

- **Ambientes virtuais tem buscado atingir níveis de realismo tão altos, que não possam ser diferenciados do mundo real**
- **Um dos fatores que determina o nível de realismo é a qualidade visual das malhas dos objetos**
- **Malhas do mundo real precisam de milhões de polígonos para serem representadas com precisão**

# Introdução

---

- **Problema:**
  - **Quanto maior o número de polígonos das malhas renderizadas, maior a demanda de memória e processamento**
- **Solução:**
  - **Simular as micro e macro deformações nas malhas, sem ter que processar a malha real dos objetos (com milhões de polígonos)**

# Objetivo

---

- **Analisar e comparar as principais técnicas existentes para simulação de malhas detalhadas**
- **7 técnicas analisadas**
- **6 técnicas comparadas**

# Motivação

---

- Existem diversas técnicas para esse propósito, sendo que poucas apresentam comparações com as demais existentes
- Não foram encontrados trabalhos que comparem as técnicas existentes
  - Muitas técnicas recentes (2004~2006)
  - Difícil comparação: diferentes APIs, linguagens de *shaders* e *hardwares*

# Metodologia

---

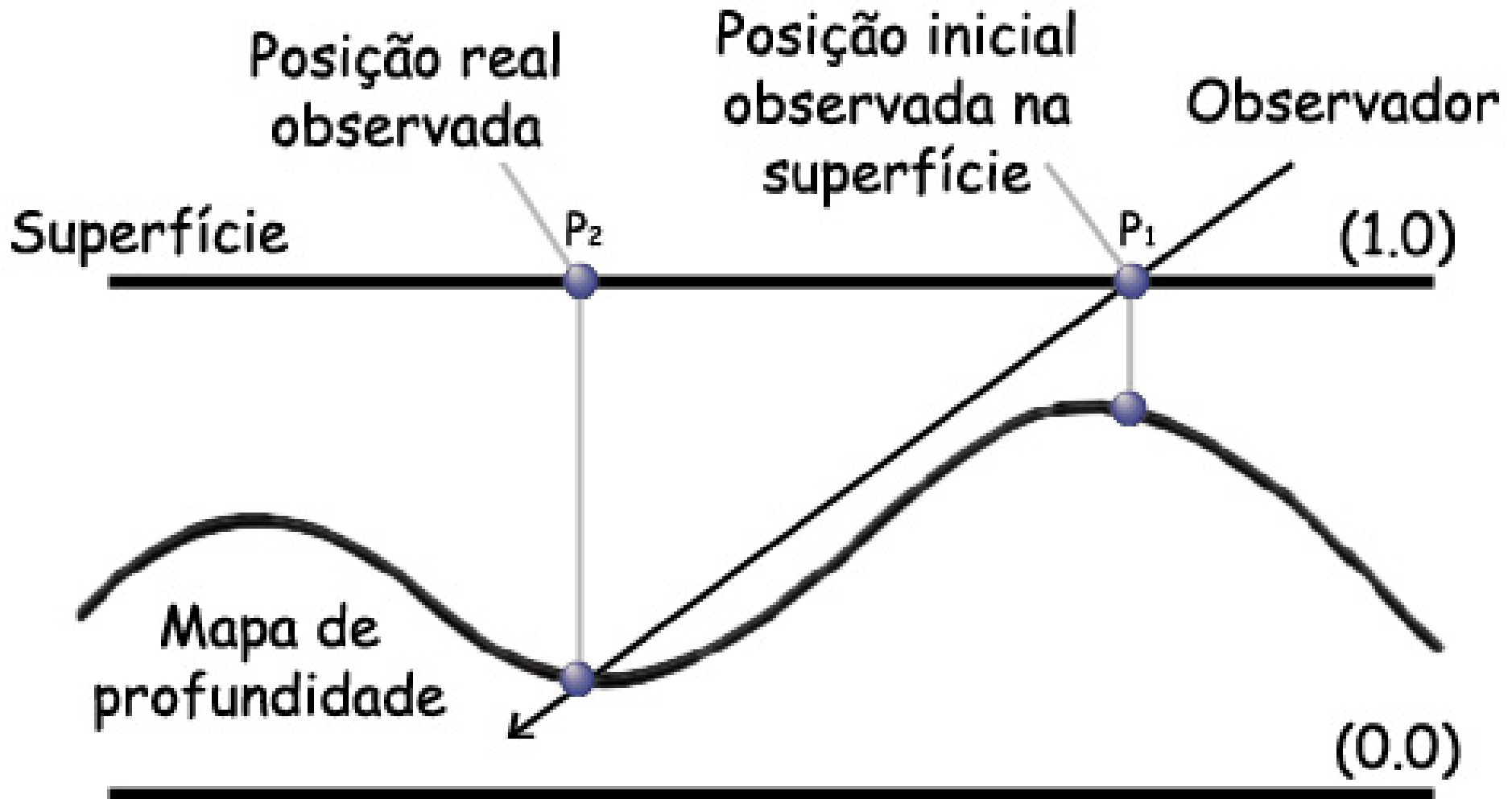
- **Metodologia de análise**
  - **Funcionamento das técnicas, efeitos visuais tratados, principais vantagens e desvantagens e ambiente para uso eficaz**
- **Metodologia de comparação**
  - **Criação de um ambiente homogêneo para testes, utilizando Direct3D 9.0c e HLSL**
  - **Comparação entre qualidade visual e desempenho das técnicas**

# Efeitos visuais desejados

---

- **Motion Parallax**
  - **Renderizar os pontos corretos observados na superfície das malhas**
- **Self-Occlusion**
  - **Uma parte da geometria pode ser ocluída por outra parte**

# Efeitos visuais desejados



# Efeitos visuais desejados

---

## ● Self-Shadow

- A geometria pode gerar sombras sobre ela mesma

## ● Silhueta

- O contorno das malhas renderizadas não deve ser o mesmo da superfície real que é renderizada

# Processo de renderização

- Teórico - Equação de renderização

$$L_0(x, \vec{w}) = L_e(x, \vec{w}) + \int_{\Omega} f_r(x, \vec{w}', \vec{w}) L_i(x, \vec{w}') (\vec{w}' \cdot \vec{n}) V(x, \vec{w}') d\vec{w}'$$

- Difícil de ser aplicada em tempo real nos *hardwares* gráficos atuais

# Processo de renderização

- Solução:
  - Reduzir a equação da renderização
- Considerações:
  - Iluminação local e sem atenuação
  - Superfícies isotrópicas e sem emissão de luz

$$L_0(x, \vec{w}) = L_e(x, \vec{w}) + \int_{\Omega} f_r(x, \vec{w}', \vec{w}) L_i(x, \vec{w}') (\vec{w}' \cdot \vec{n}) V(x, \vec{w}') d\vec{w}'$$

Similar a equação de Phong

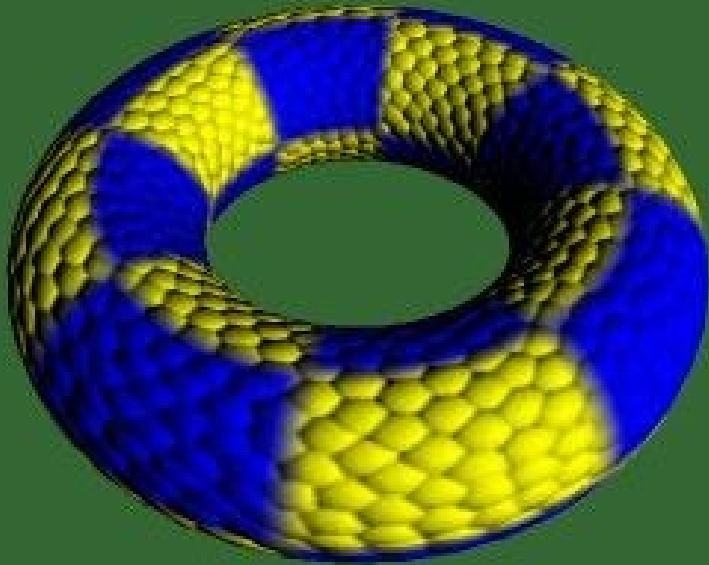
# Processo de renderização

---

- **Resultado**
  - **Cor final de cada ponto renderizado na superfície depende da normal da superfície e do vetor de iluminação**

# Bump Mapping

---



**Superfície com  
Bump Mapping**



**Superfície real**

# Bump Mapping

---

- **O Bom**
  - **Aumenta os detalhes das superfícies sem sofrer grande redução no desempenho**
  - **Facilmente implementado nas GPUs e integrado a pipelines existentes**
- **O Ruim**
  - **Não consegue representar grandes detalhes na superfícies**
  - **Não trata nenhum dos efeitos visuais desejados**

# Parallax Mapping



**Superfície com  
Bump Mapping**

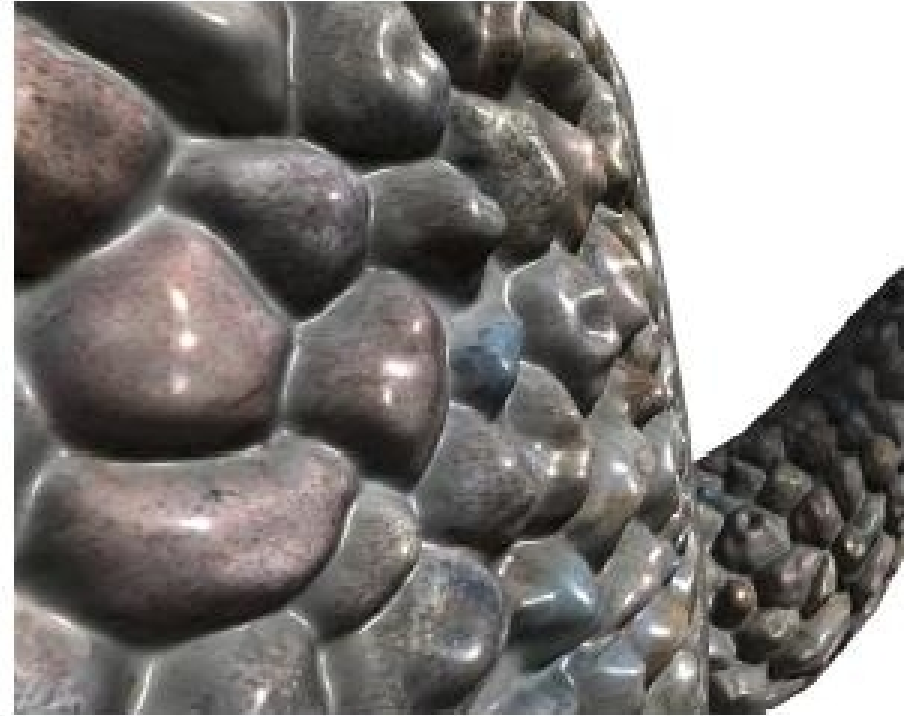


**Superfície com  
Parallax Mapping**

# Parallax Mapping

- **O Bom**
  - **Aumenta os detalhes das superfícies sem sofrer grande redução no desempenho**
  - **Facilmente implementado em conjunto com a técnica de Bump Mapping**
- **O Ruim**
  - **É uma aproximação do efeito real**
  - **Qualidade visual ruim quando utiliza mapas de profundidade genéricos**
  - **Não trata os efeitos de *self-occlusion*, *self-shadow* e *silhuetas***

# Relief Mapping



**Superfícies lisas renderizadas com  
Relief Mapping**

# Relief Mapping

---

- **O Bom**
  - **Consegue renderizar superfícies detalhadas com precisão e alto nível de realismo**
  - **Trata os efeitos de *motion parallax*, *self-occlusion* e *self-shadow***
- **O Ruim**
  - **Necessita de *hardwares* gráficos modernos**
  - **Demanda alto processamento**
  - **Pode gerar artefatos nas imagens renderizadas**

# Parallax Occlusion Mapping



**Superfície real**  
**1.5 milhões de triângulos**



**Superfície com POM**  
**1.100 triângulos**

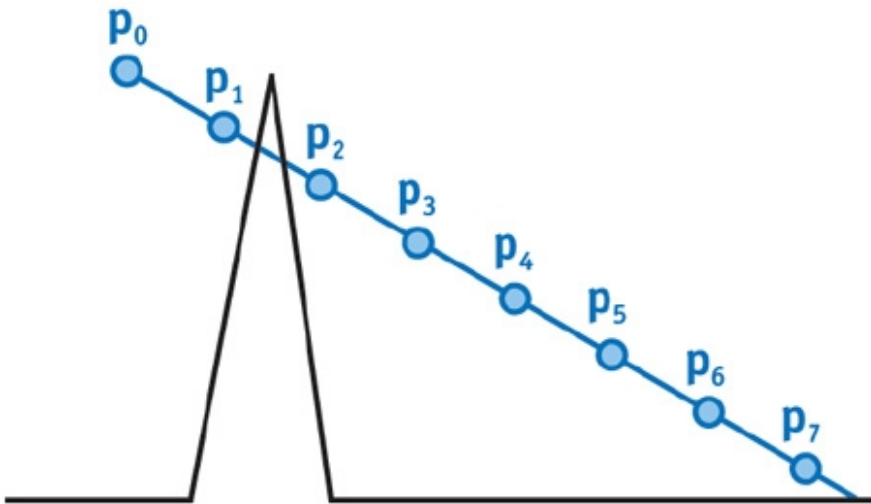
# Parallax Occlusion Mapping

- O Bom
- Consegue renderizar superfícies detalhadas com precisão e alto nível de realismo
- Trata os efeitos de *motion parallax*, *self-occlusion* e *self-shadow* (sombras suaves)
- Pode diminuir o serrilhamento e artefatos gerados na imagem

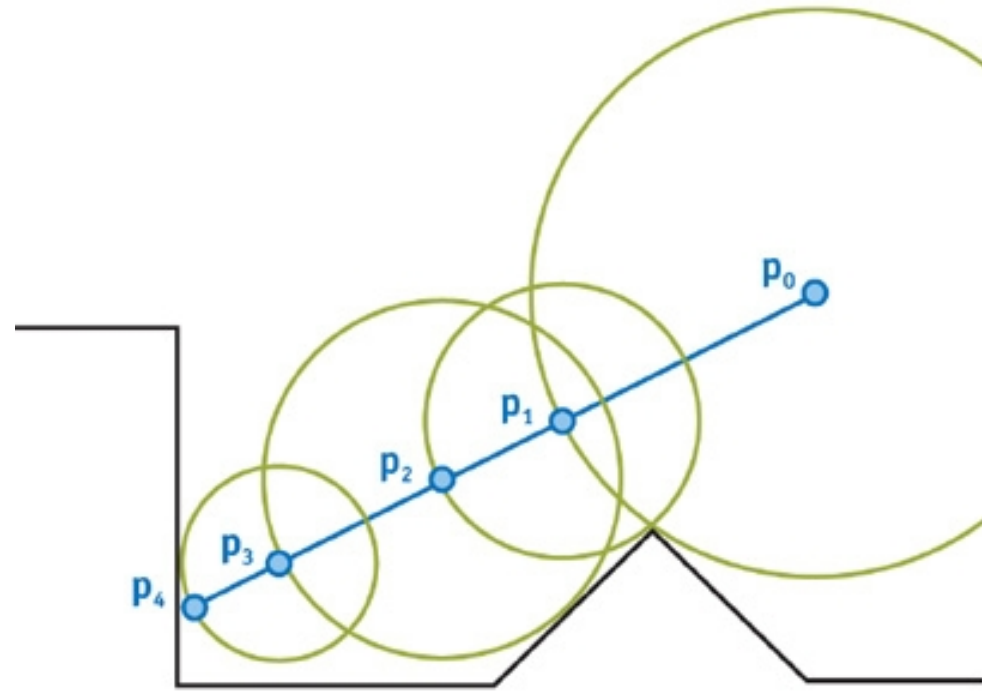
# Parallax Occlusion Mapping

- O Ruim
- Necessita de *hardwares* gráficos modernos
- Demanda alto processamento, muito superior a técnica de Relief Mapping
- Necessita de um grande número de iterações para reduzir o serrilhamento nas superfícies

# Sphere Tracing



**Algoritmo padrão de  
traçado de raios  
(Pode perder colisões)**



**Algoritmo de Sphere  
Tracing  
Dificulta a perda de  
colisões**

# Sphere Tracing

---

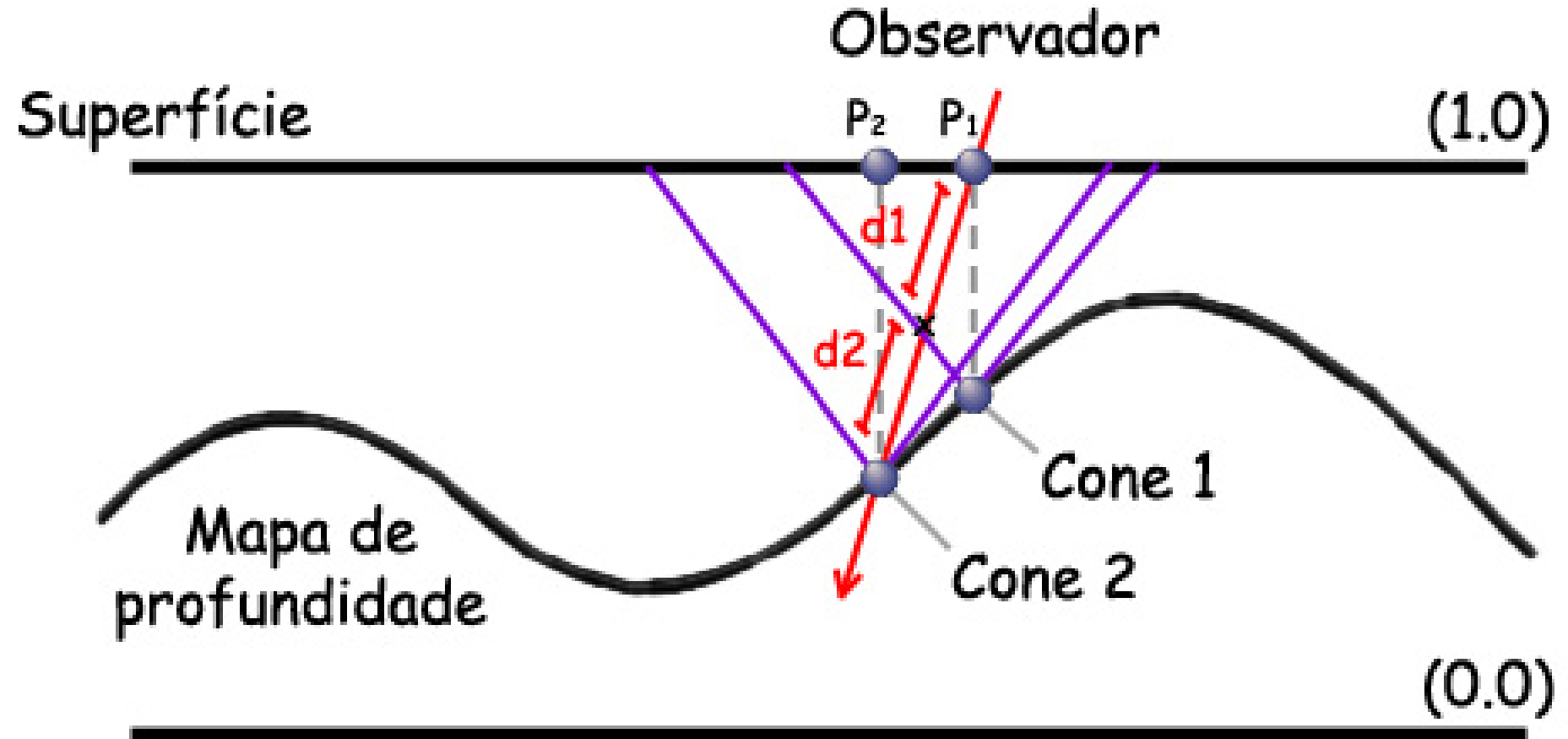
- O Bom
- Consegue renderizar superfícies detalhadas com precisão e alto nível de realismo
- Trata os efeitos de *motion parallax*, *self-occlusion* e *self-shadow*\*
- Não gera artefatos

# Sphere Tracing

---

- **O Ruim**
- **Necessita de *hardwares* gráficos modernos**
- **Demanda alto processamento**
- **Demanda alto consumo de memória**
- **Pode gerar pequenas deformações ou reduzir o nível de extrusão das superfícies renderizadas**

# Cone Step Mapping



Algoritmo de Cone Step Mapping  
Dificulta a perda de colisões

# Cone Step Mapping

---

- O Bom
- Consegue renderizar superfícies detalhadas com precisão e alto nível de realismo
- Trata os efeitos de *motion parallax*, *self-occlusion* e *self-shadow*\*
- Não gera artefatos

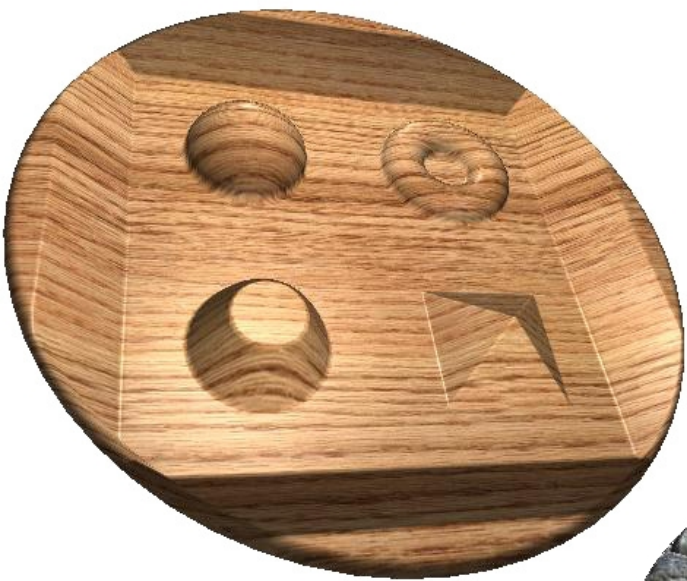
# Cone Step Mapping

---

- **O Ruim**
  - **Necessita de *hardwares* gráficos modernos**
  - **Demanda alto processamento**
  - **Pode gerar pequenas deformações ou reduzir o nível de extrusão das superfícies renderizadas**

# Comparação entre as técnicas

- As técnicas foram comparadas quando a sua renderização em 5 diferentes tipos de superfícies
- Para tipo de superfície foram utilizados dois pontos de vista distintos
  - Distância média e ângulo de observação próximo de 0 graus
  - Distância pequena e ângulo de observação próximo de 90 graus



# Comparação entre imagens

**Video do ambiente de testes criado**

# Comparação de desempenho

|                   | 1º Sup. | 2º Sup. | 3º Sup. | 4º Sup. | 5º Sup. |
|-------------------|---------|---------|---------|---------|---------|
| Bump Mapping      | 805     | 818     | 809     | 810     | 817     |
| Parallax Mapping  | 805     | 809     | 803     | 804     | 788     |
| Relief Mapping    | 300     | 269     | 280     | 246     | 180     |
| POM               | 35      | 29      | 36      | 32      | 23      |
| Cone Step Mapping | 208     | 199     | 204     | 180     | 139     |
| Sphere Tracing    | 318     | 268     | 293     | 219     | 166     |

**Comparação entre o desempenho das técnicas para as 5 superfícies avaliadas, observadas pelo primeiro ponto de vista**

# Conclusão

---

- **Nenhuma das técnicas apresenta uma solução genérica para simulação de malhas detalhadas**
- **Cada uma das técnicas possui uma diferente relação entre qualidade visual e desempenho**
- **Além de poderem tratar diferentes efeitos visuais**

# Perguntas ?

---

**Bruno Pereira Evangelista**

[bpevangelista@gmail.com](mailto:bpevangelista@gmail.com)

**Orientador:**

**Marcelo Souza Nery**

[msnery@pucminas.br](mailto:msnery@pucminas.br)

**"De fato, que aproveitará ao homem ganhar o mundo inteiro mas perder sua alma?"**

**Mateus 16, 26**