

# Técnicas de oclusão aplicadas a visualização de silhuetas

**Sinésio Pesco**



**Laboratório Matmidia**

**Departamento de Matemática**

**Pontifícia Universidade Católica do Rio de Janeiro**

# Introdução

## Motivação

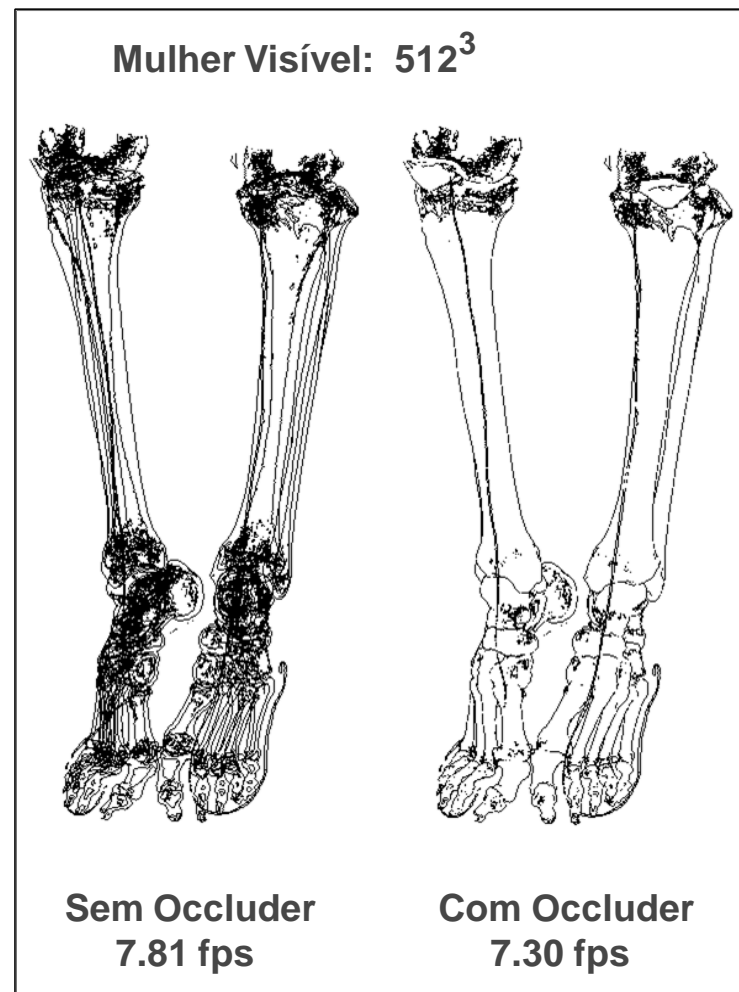
- Silhuetas são um importante instrumento para análise e visualização de dados.

## Desafios

- Linhas não-visíveis podem comprometer a interpretação.
- A remoção das linhas não-visíveis aumenta o tempo de visualização.

## Objetivo

- Acelerar o tempo de remoção de linhas não-visíveis.

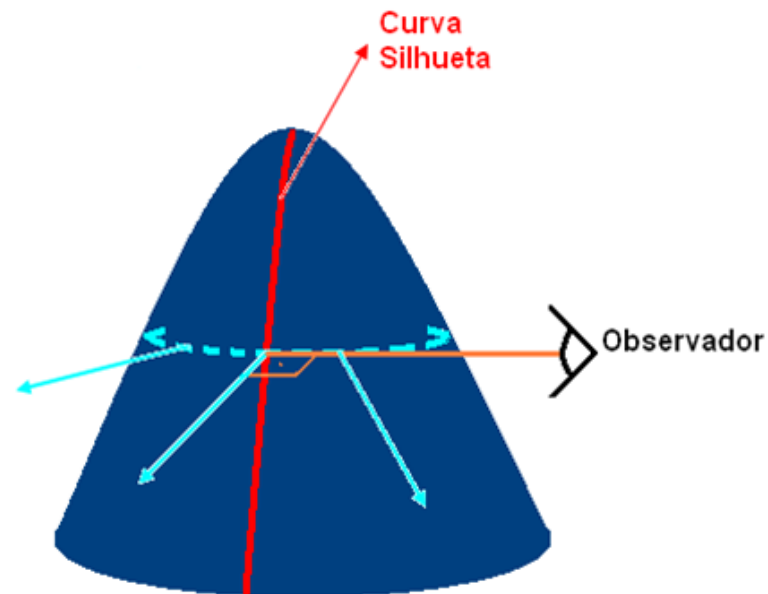


- **Silhueta**
- **Oclusão Implícita**
- **Resultados**
- **Conclusão**

# Silhueta

## Definição

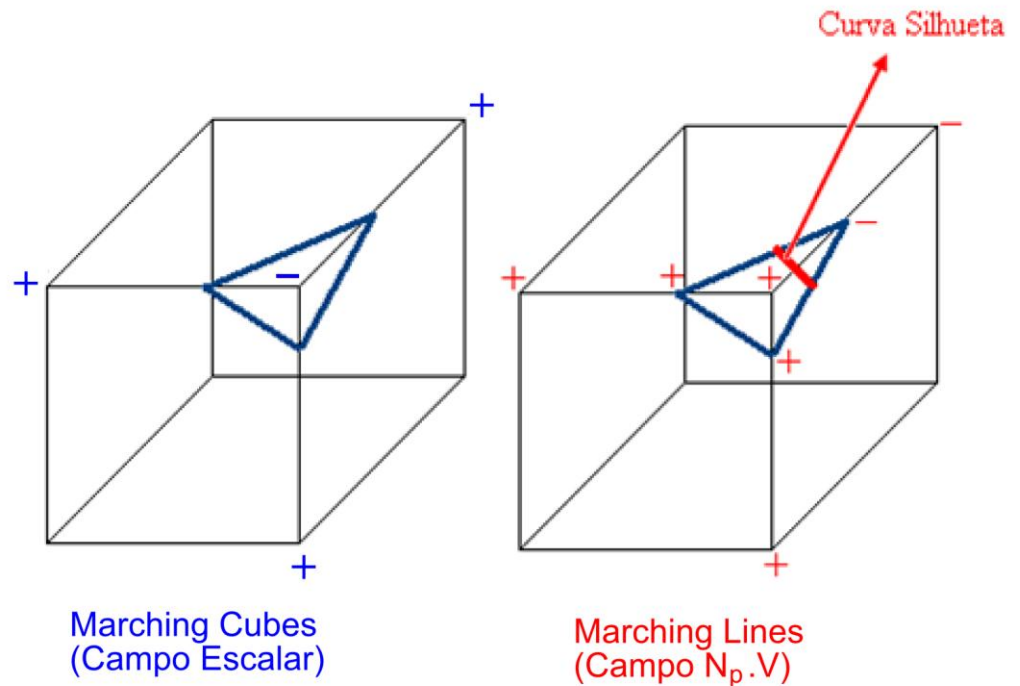
- A curva silhueta de uma superfície  $S$  é formada pelos pontos  $p$  pertencentes a  $S$  tal que  $N_p \cdot V = 0$ , onde  $N_p$  é o vetor normal à superfície e  $V$  o vetor do ponto de visão.



# Silhueta

## Extração da Curva Silhueta

- Dados volumétricos regulares.
- Marching Lines  
[Thirion, J.P., Gourdon, A., *The 3d marching lines algorithm*, 1996].



# Silhueta

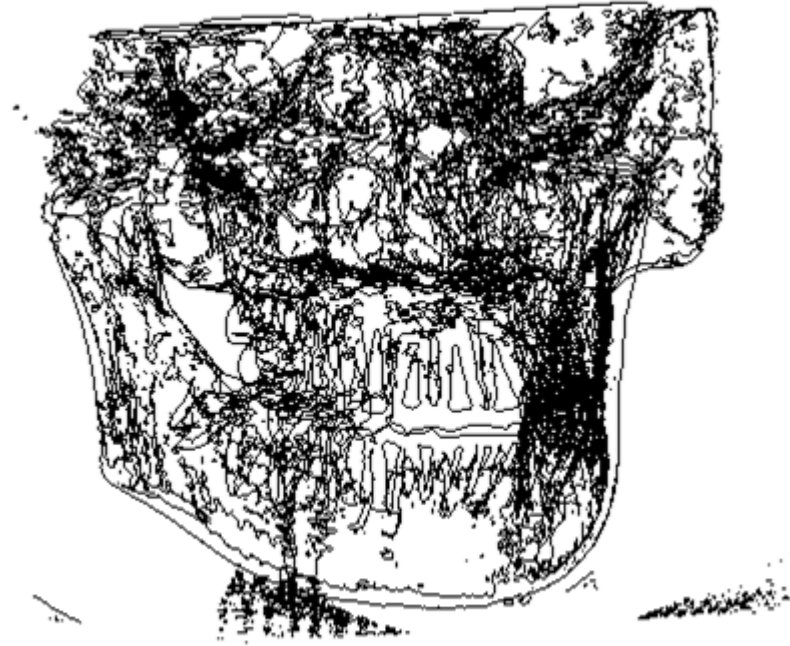
## Extração da Curva Silhueta

- Utiliza uma octree para armazenar a informação min-max por nó.  
[Wilhelms J., Van Gelder A., *Octrees for Faster Isosurface Generation*, 1992].
- Percorre a curva silhueta a partir de um nó inicial (semente).  
[Burns, M., Klawe, J., Rusinkiewicz, S., Finkelstein, A., DeCarlo, D., *Line Drawings from Volume Data*, 2005]
- Utilizamos a estratégia:
  - 1) Inicialmente buscar as sementes nos nós classificados como zero.
  - 2) Nos passos seguintes, buscar sementes nos nós encontrados ou seus vizinhos.

# Silhueta

## Extração da Curva Silhueta

Skull (512 x 512 x 512)



Passo 1: 0,625 seg  
Passo 2: 0,141 seg

Bonsai (256 x 256 x 256)



Passo 1: 0,251 seg  
Passo 2: 0,039 seg

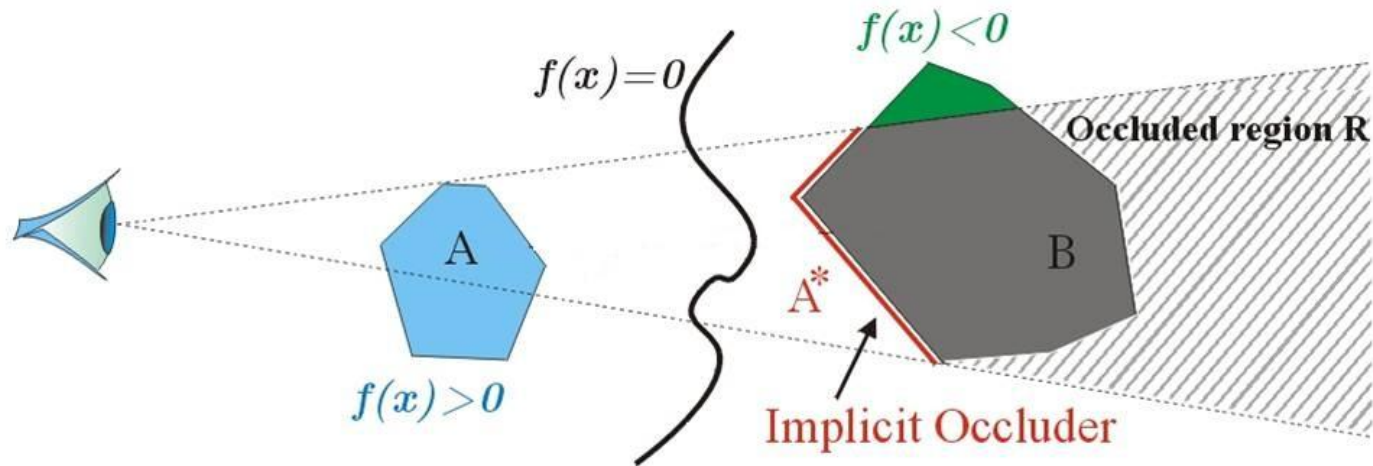
# Oclusão Implícita

## Introdução

- Oclusão Implícita  
[Pesco, S., Lindstrom, P., Pascucci, V., Silva, C., *Implicit Occluders*, 2004].
- Técnica de visibilidade para otimizar o cálculo e rendering de isosuperfícies.
- Considere  $f:R^3 \rightarrow R$  contínua. A idéia principal consiste em explorar a continuidade de  $f$  para encontrar occluders (regiões oclusas) sem a necessidade de calcular explicitamente a isosuperfície.



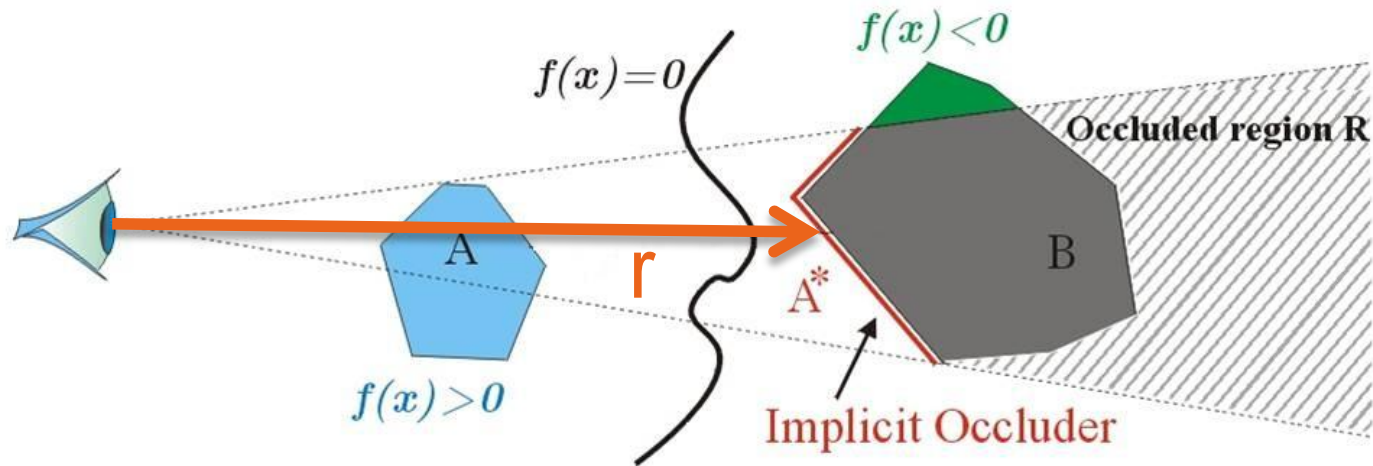
# Oclusão Implícita



Considere:

- $f$  um campo escalar contínuo definido em um domínio convexo  $D$ .
- região A:  $f(x)$  é positiva.
- região B:  $f(x)$  é negativa.
- projeção  $A^*$  da região A sobre o bordo de B.
- um particular ponto de visão.

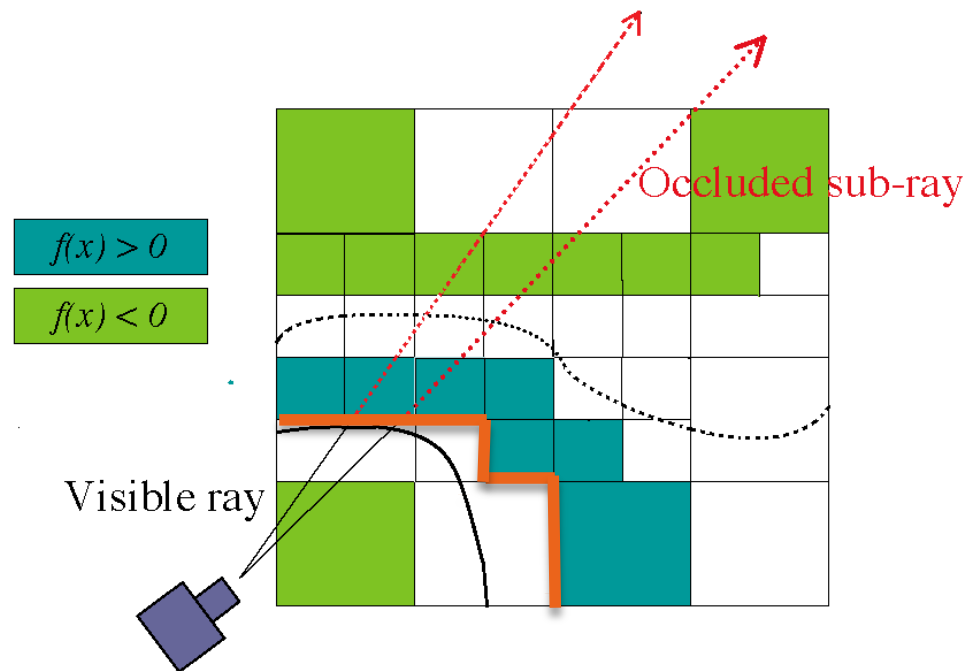
# Oclusão Implícita



- Qualquer segmento  $r$  conectando o ponto de visão com  $A^*$  terá interseção com a isosuperfície  $f^{-1}(0)$ .
- Portanto a região  $R$ , atrás de  $A^*$ , não é visível e pode ser utilizada como um occluder.

# Oclusão Implícita

- Utiliza uma octree para armazenar a informação min-max por nó.  
[Wilhelms J., Van Gelder A., *Octrees for Faster Isosurface Generation*, 1992].
- Determina a primeira troca de sinal, utilizando o stencil buffer e o depth buffer.
- A implementação é baseada em uma estratégia de dois passos, que trata primeiro os nós negativos e no segundo passo os nós positivos.



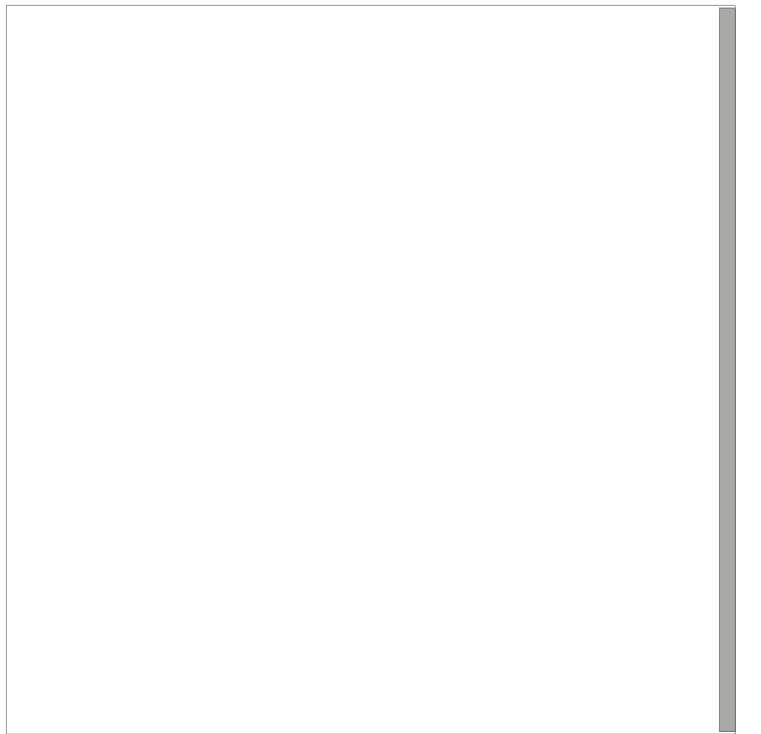
# Oclusão Implícita

## Algoritmo

- Zerar os buffers

```
glClearDepth(1);
```

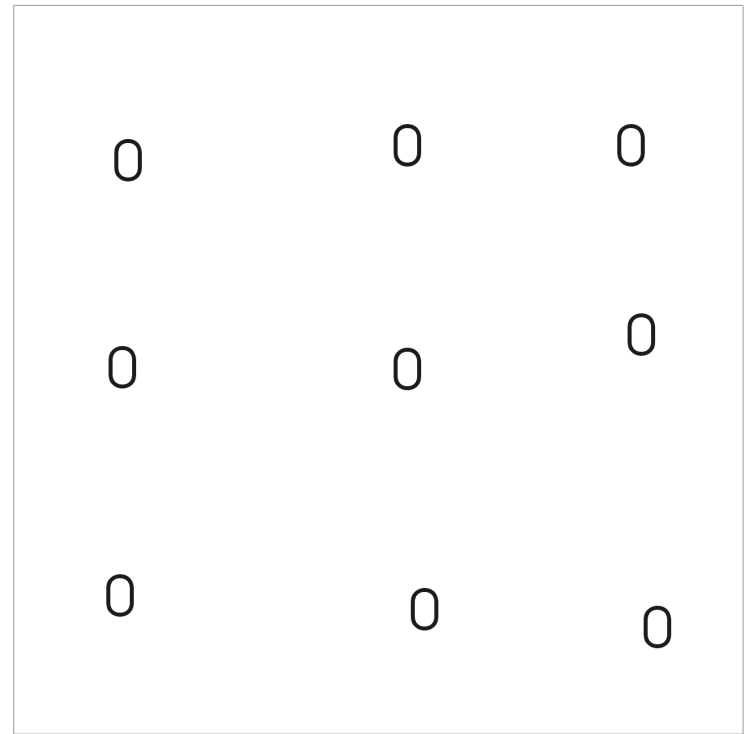
Depth Buffer



→  
Front to Back

```
glClearStencil(0);
```

Stencil Buffer



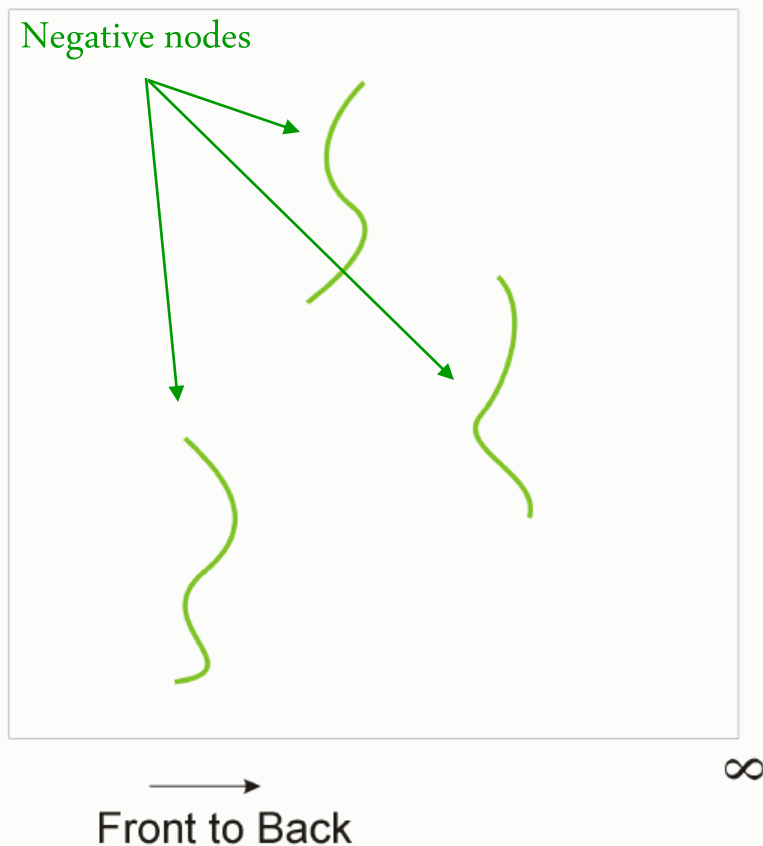
# Oclusão Implícita

## Algoritmo

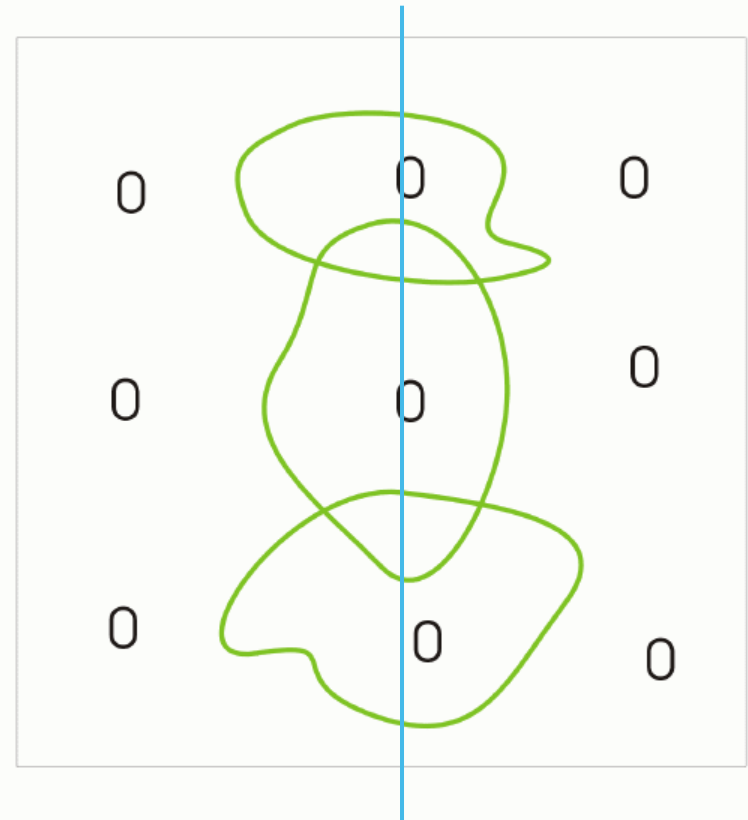
Primeiro passo:

- Os nós negativos serão renderizados.

Depth Buffer



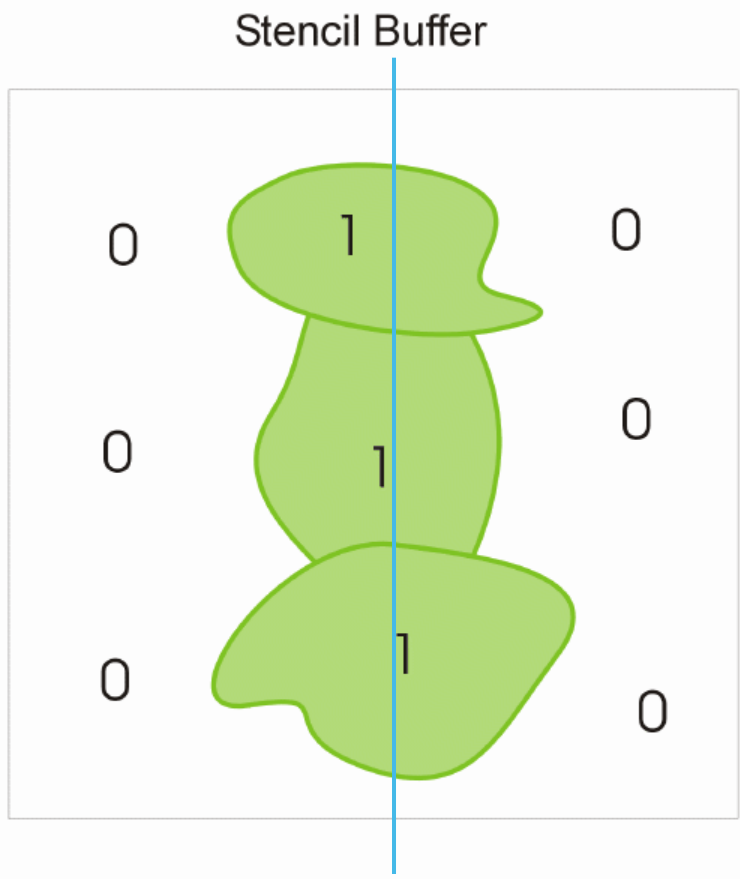
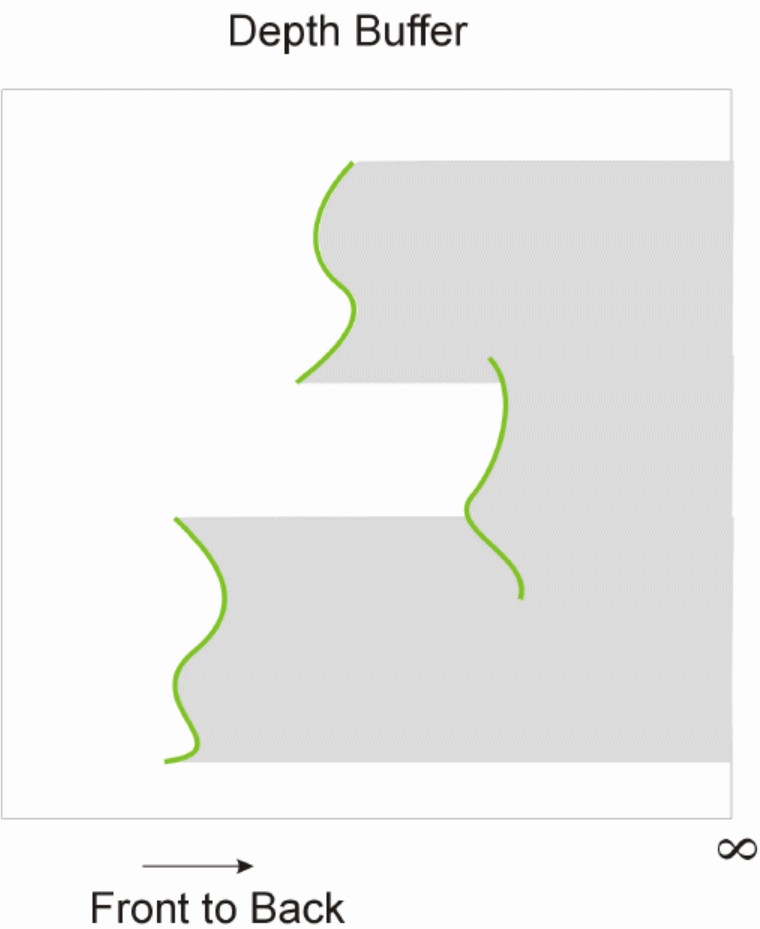
Stencil Buffer



# Oclusão Implícita

## Algoritmo

Primeiro passo:

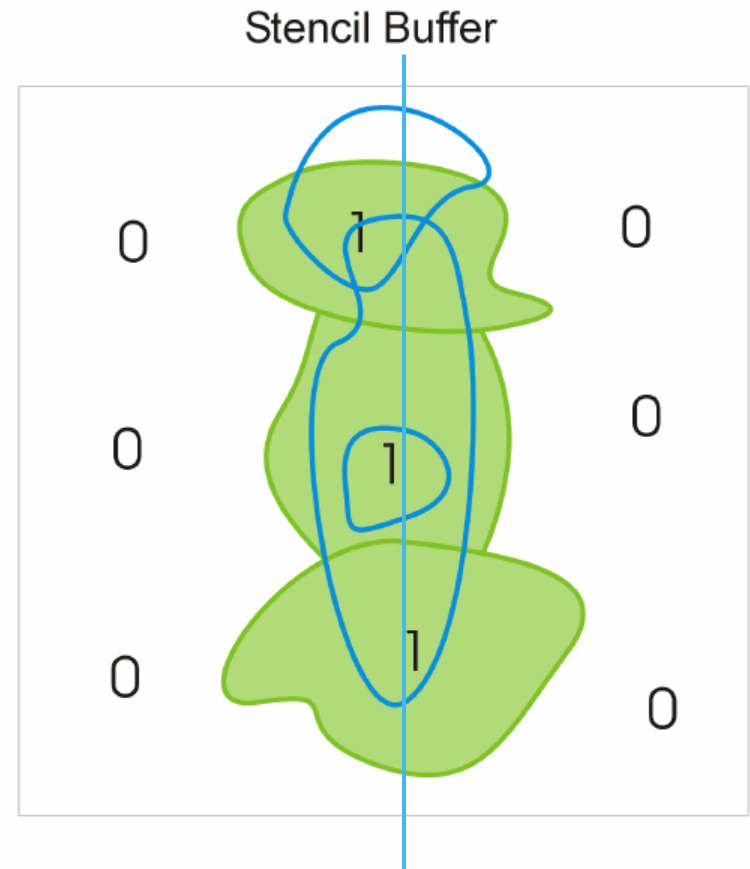
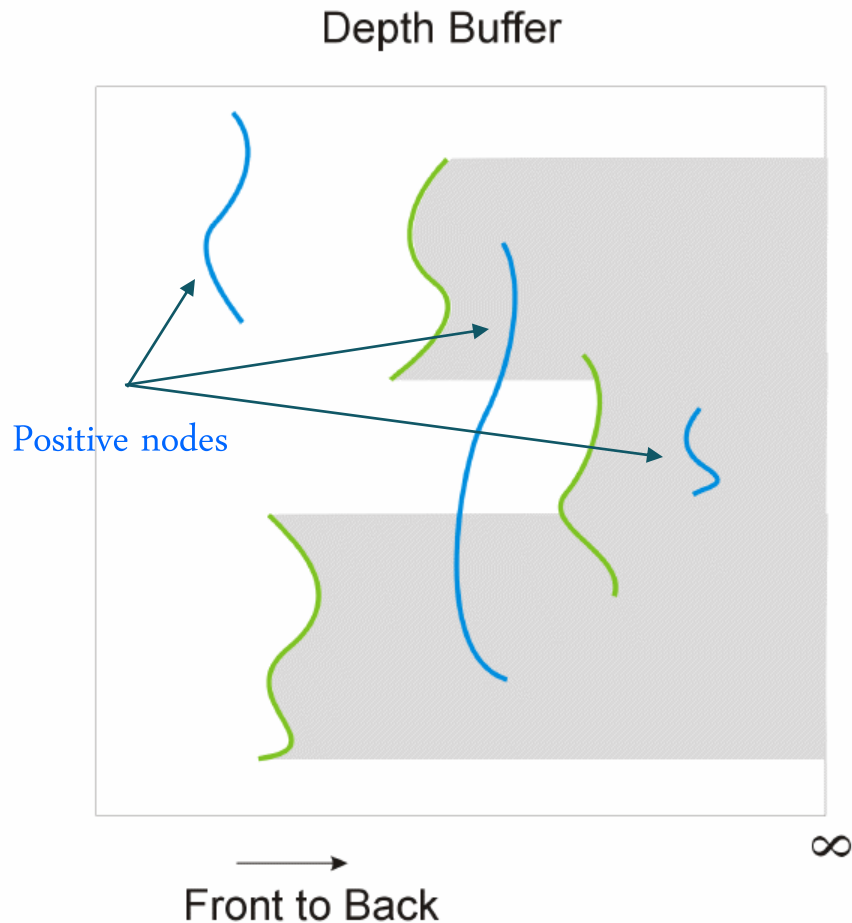


# Oclusão Implícita

## Algoritmo

Segundo passo:

- Os nós positivos serão renderizados.

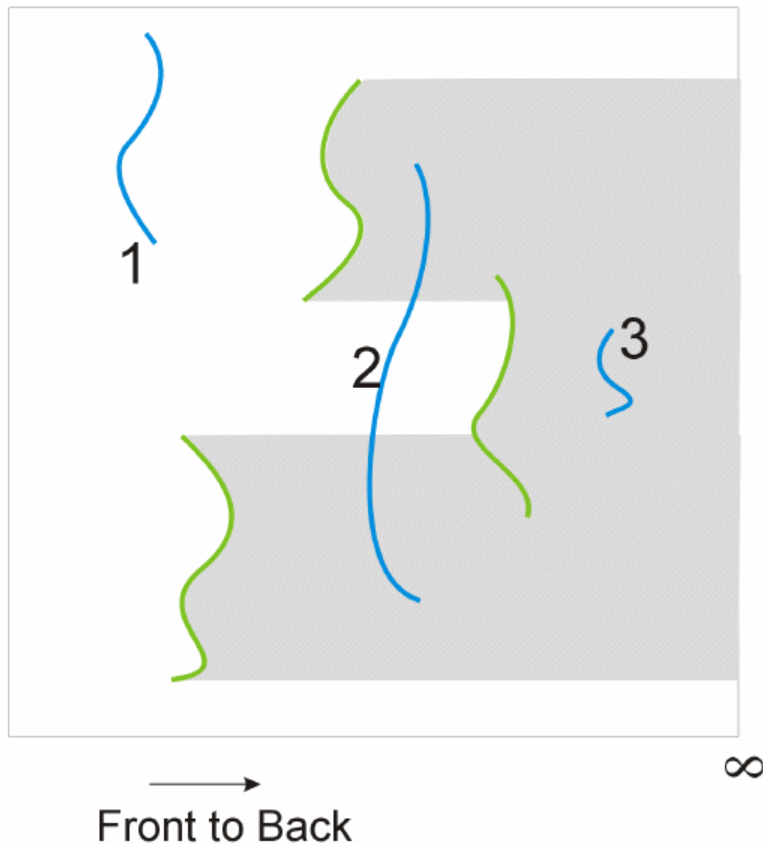


# Oclusão Implícita

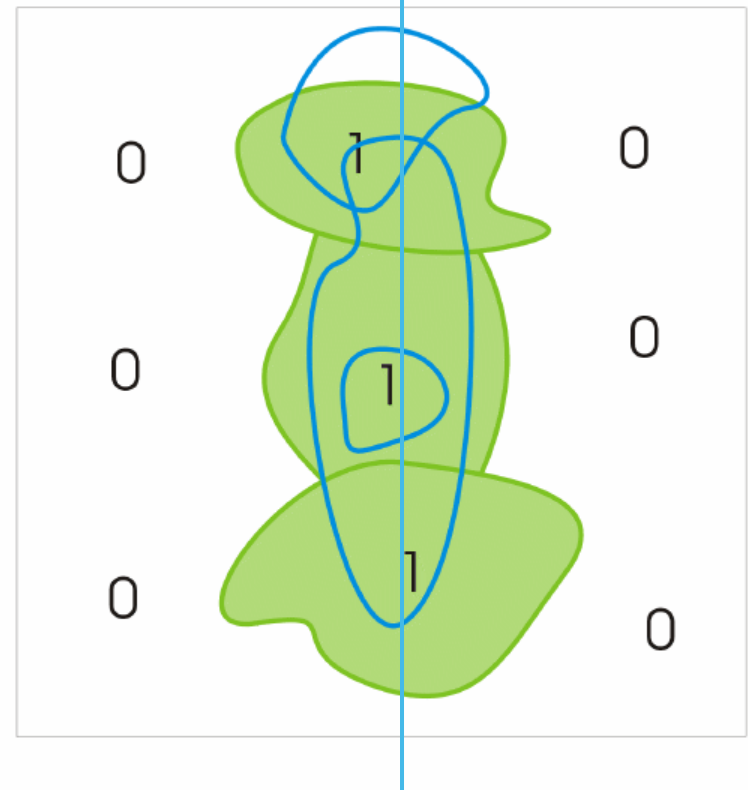
## Segundo passo:

- Nessa etapa a renderização será feita de frente para trás.
- O Depth buffer é setado para a maior profundidade.
- O stencil buffer assegura que a profundidade seja alterada uma unica vez.

Depth Buffer



Stencil Buffer

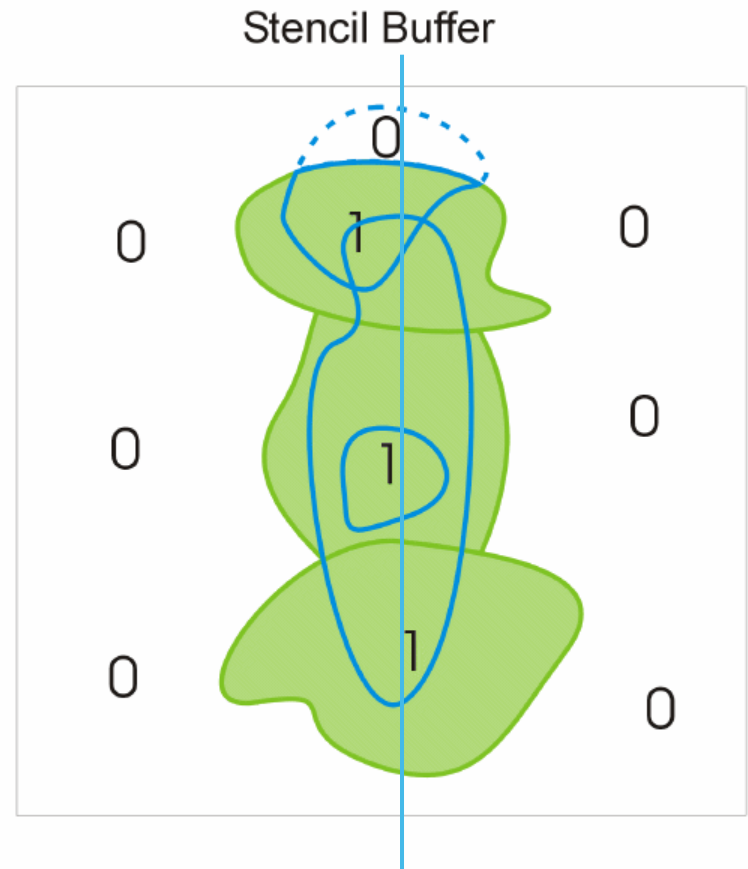
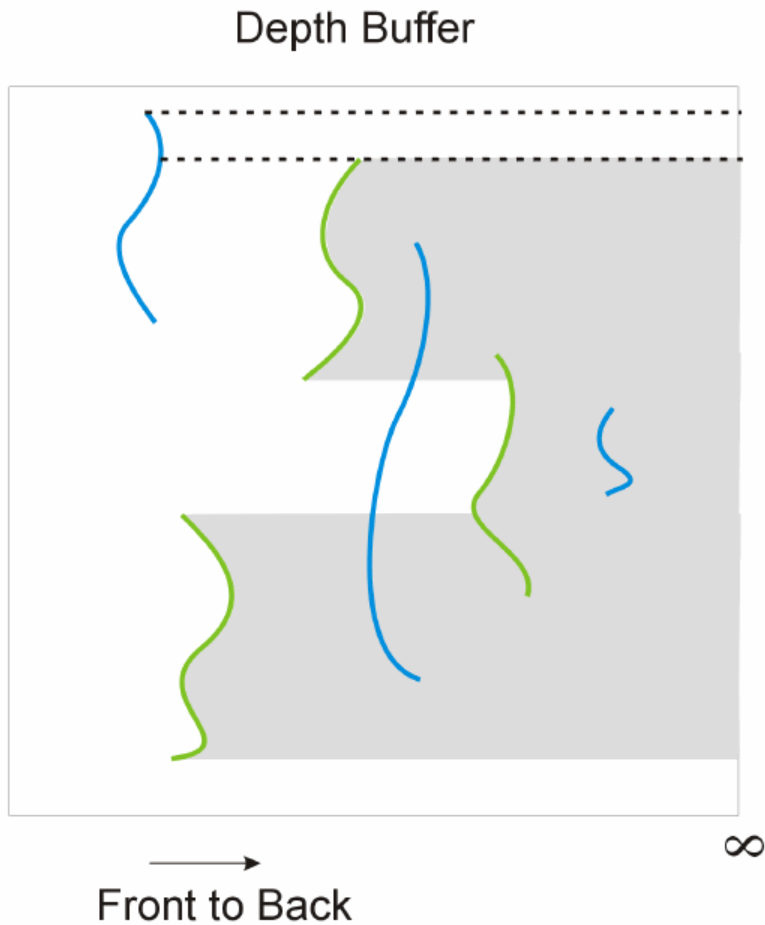




# Oclusão Implícita

## Algoritmo

Segundo passo:

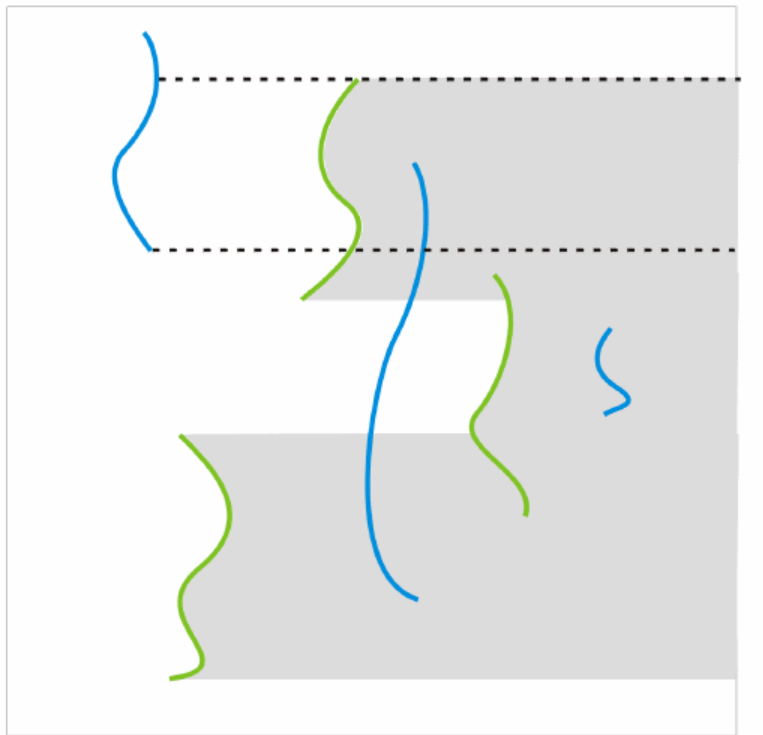


# Oclusão Implícita

## Algoritmo

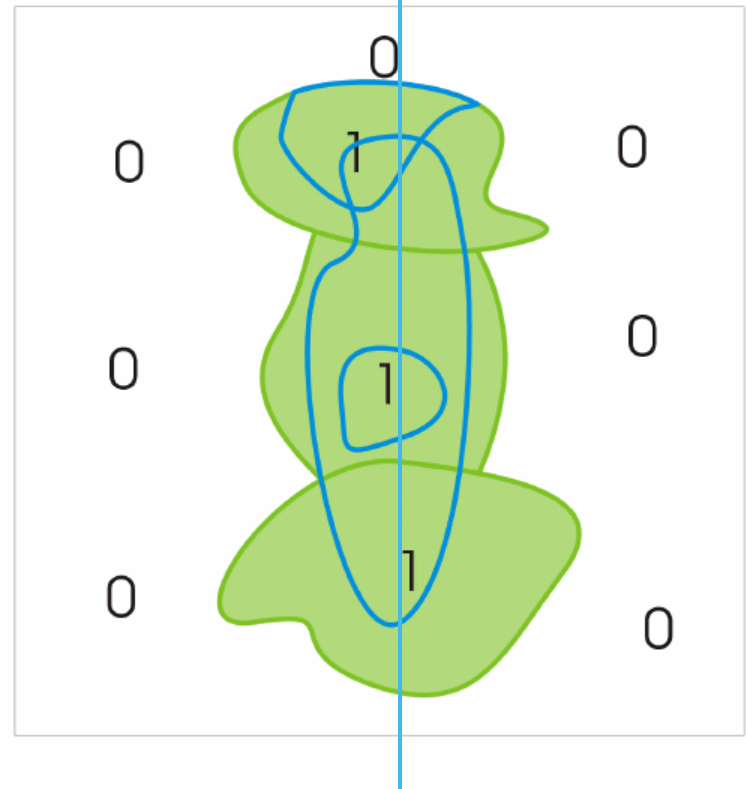
Segundo passo:

Depth Buffer



→  
Front to Back

Stencil Buffer

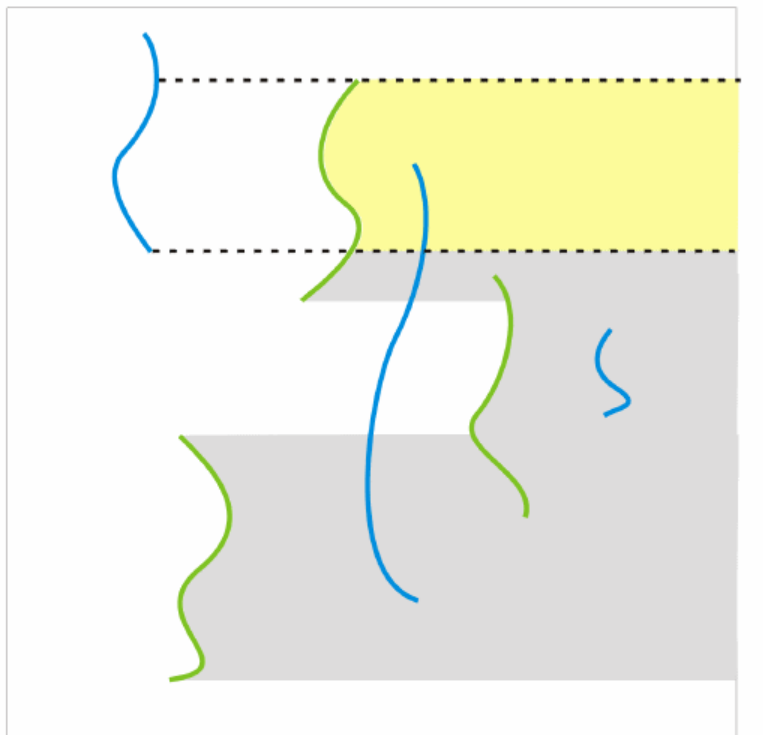


# Oclusão Implícita

## Algoritmo

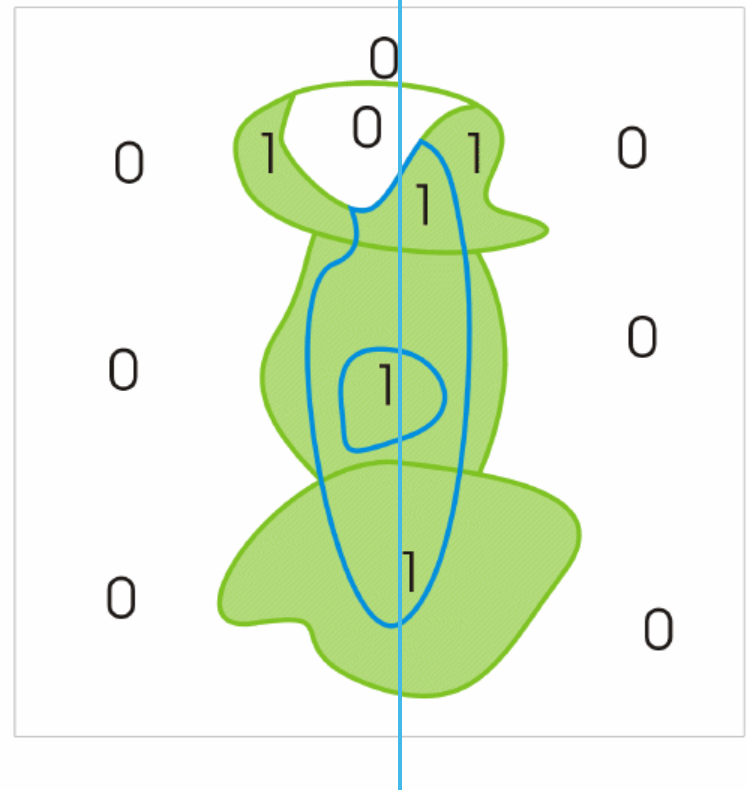
Segundo passo:

Depth Buffer



→  
Front to Back

Stencil Buffer

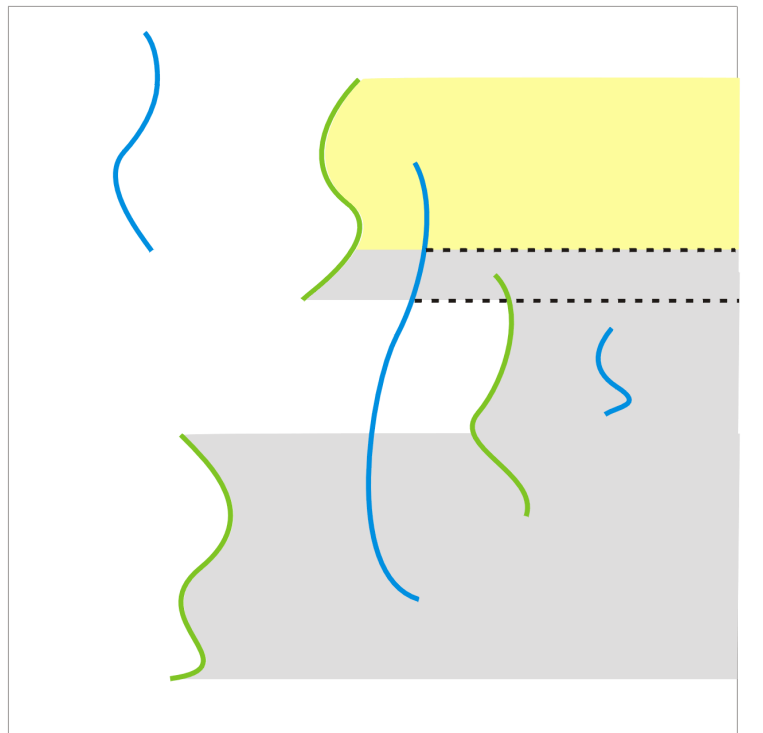


# Oclusão Implícita

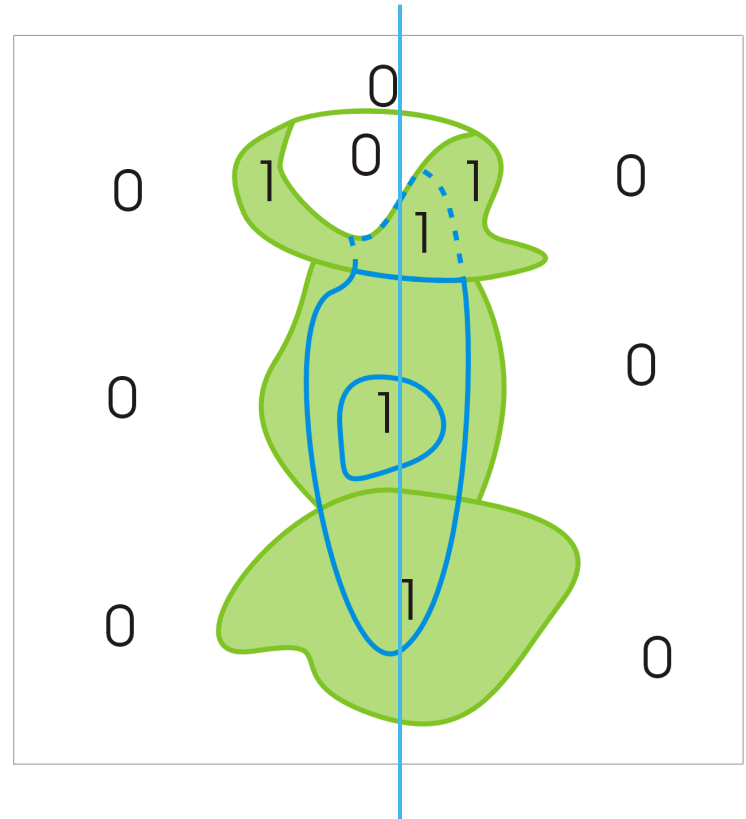
## Algoritmo

Segundo passo:

Depth Buffer



Stencil Buffer



→  
Front to Back

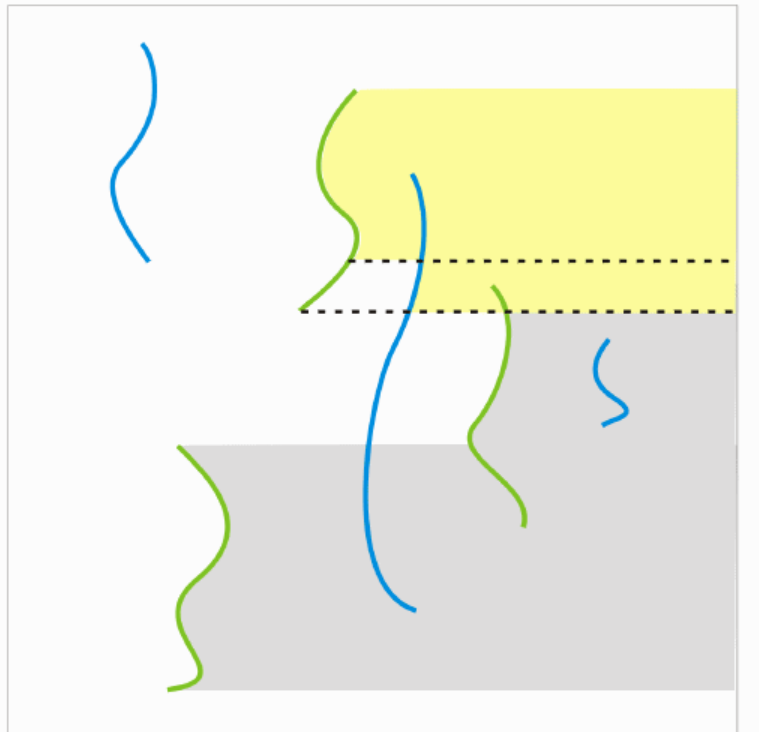
$\infty$

# Oclusão Implícita

## Algoritmo

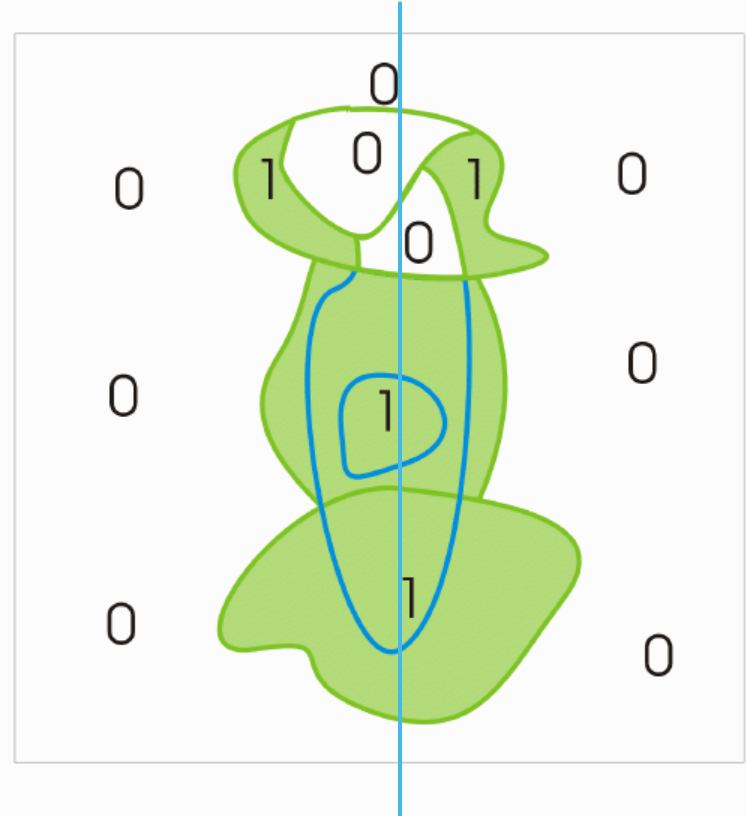
Segundo passo:

Depth Buffer



→  
Front to Back

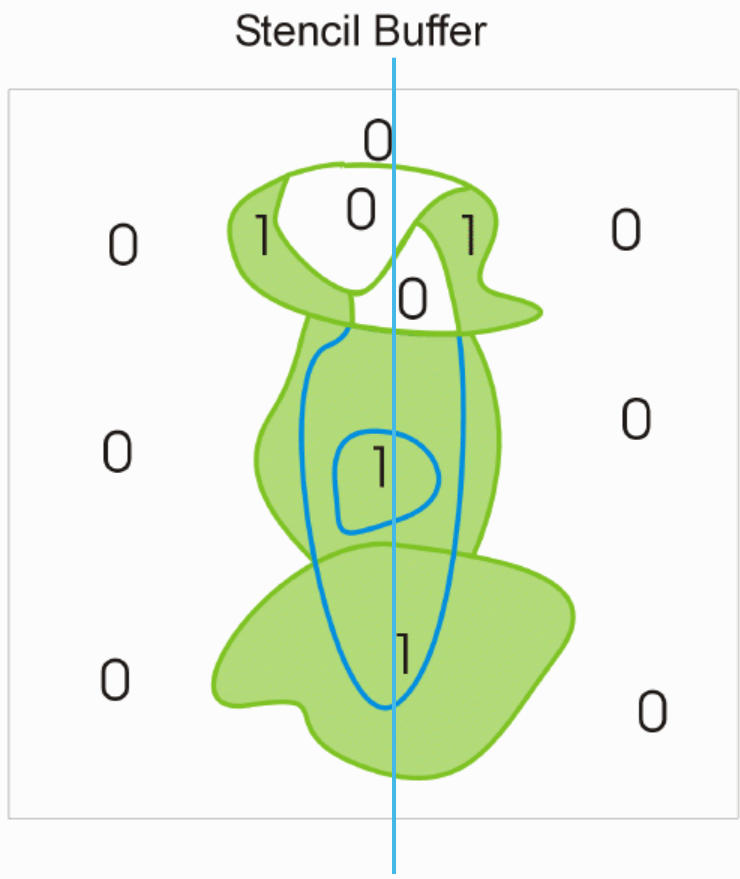
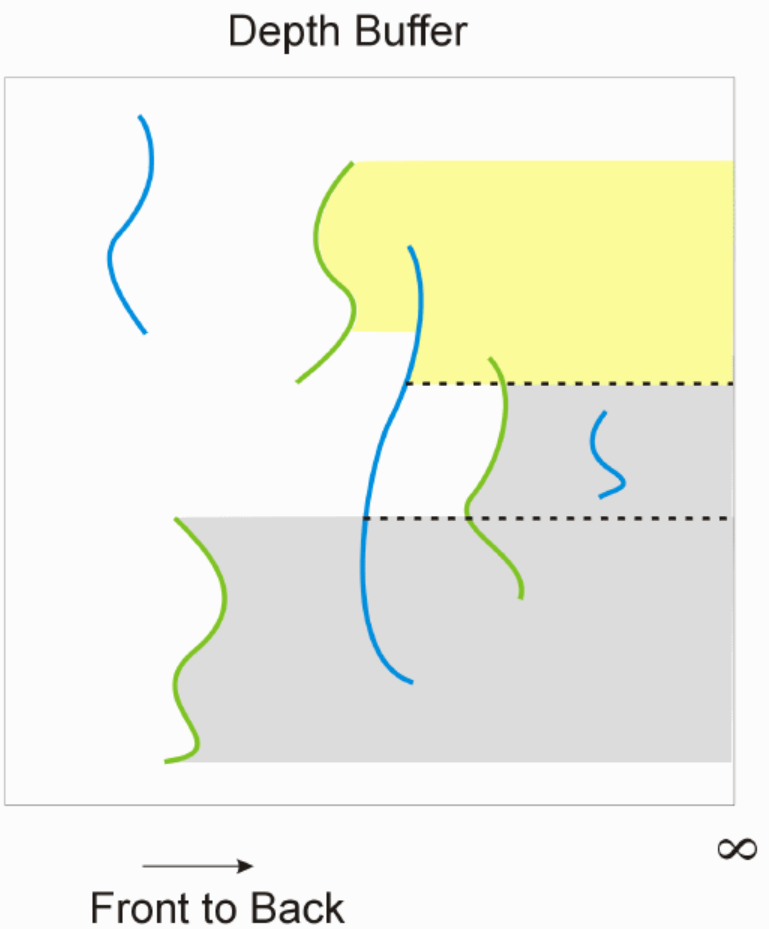
Stencil Buffer



# Oclusão Implícita

## Algoritmo

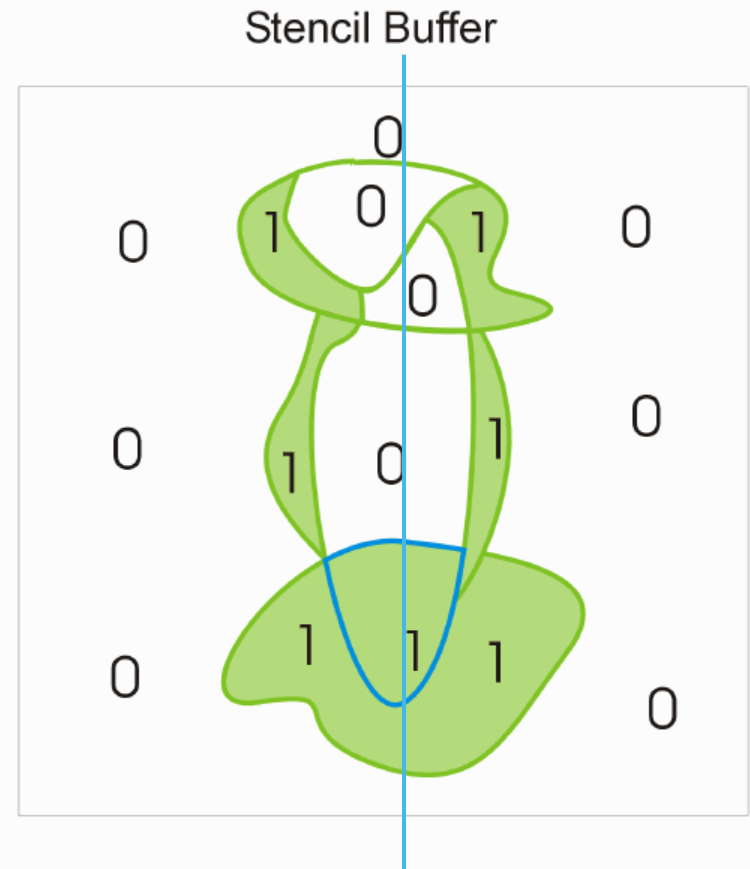
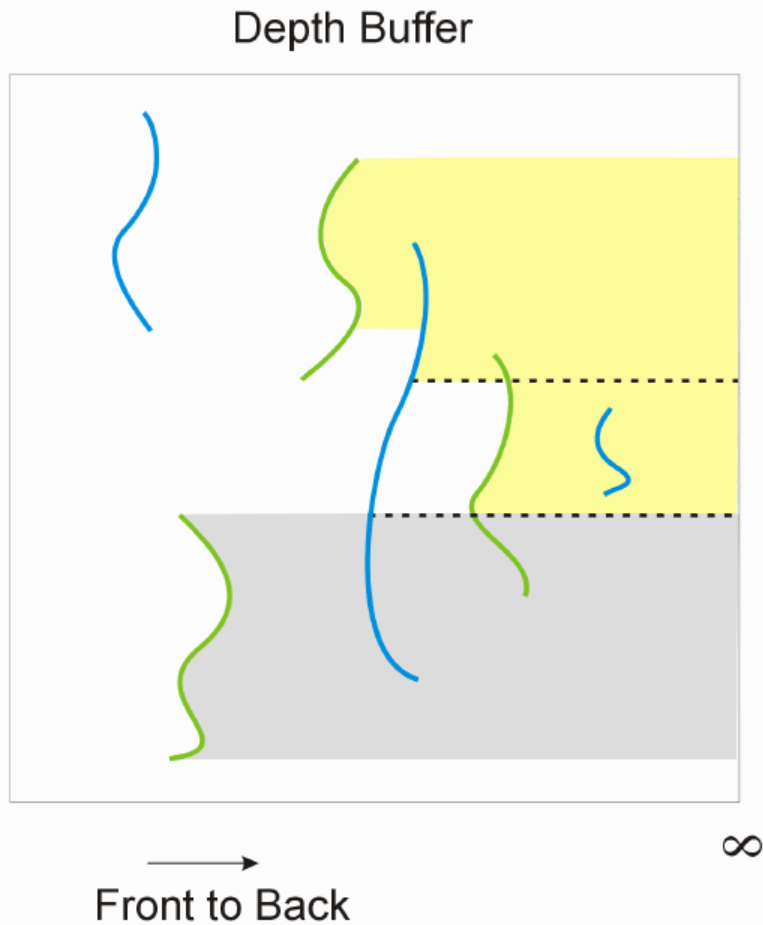
Segundo passo:



# Oclusão Implícita

## Algoritmo

Segundo passo:

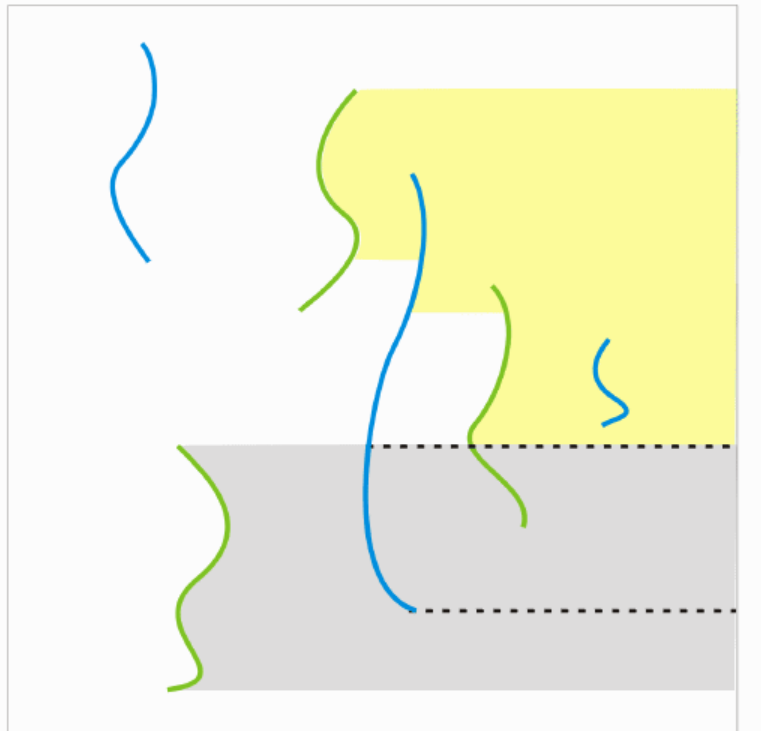


# Oclusão Implícita

## Algoritmo

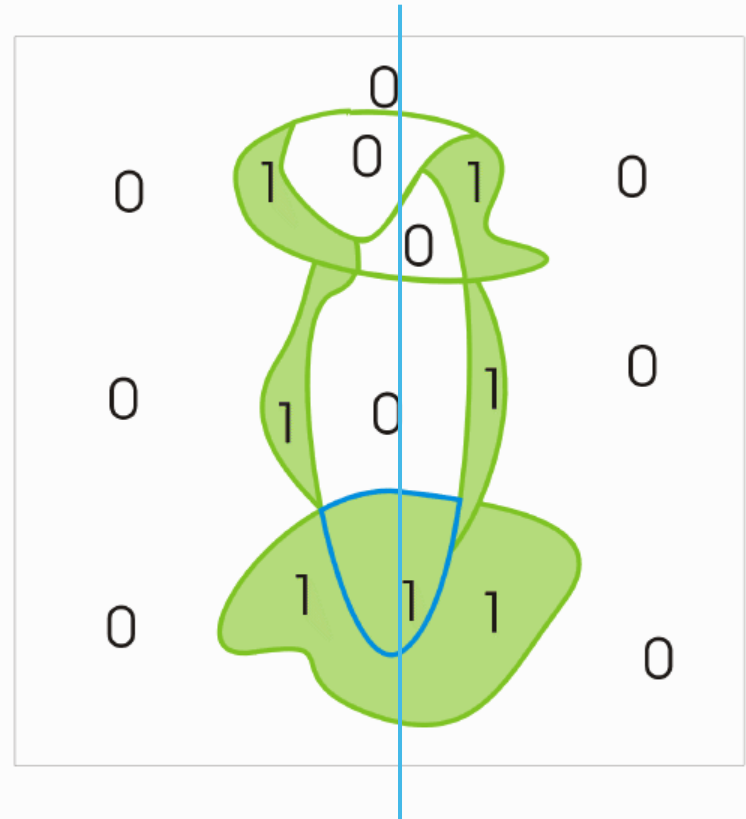
Segundo passo:

Depth Buffer



→  
Front to Back

Stencil Buffer



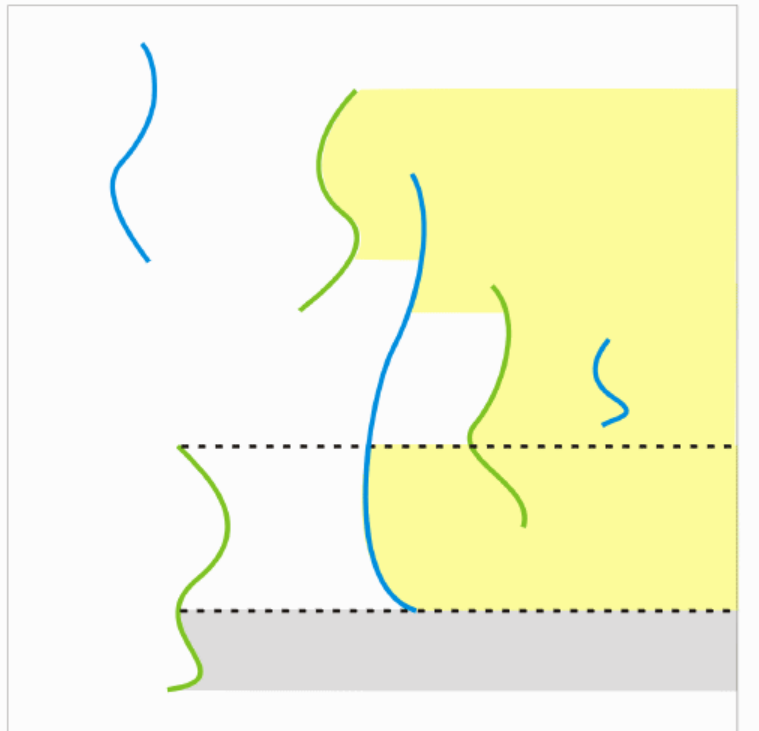


# Oclusão Implícita

## Algoritmo

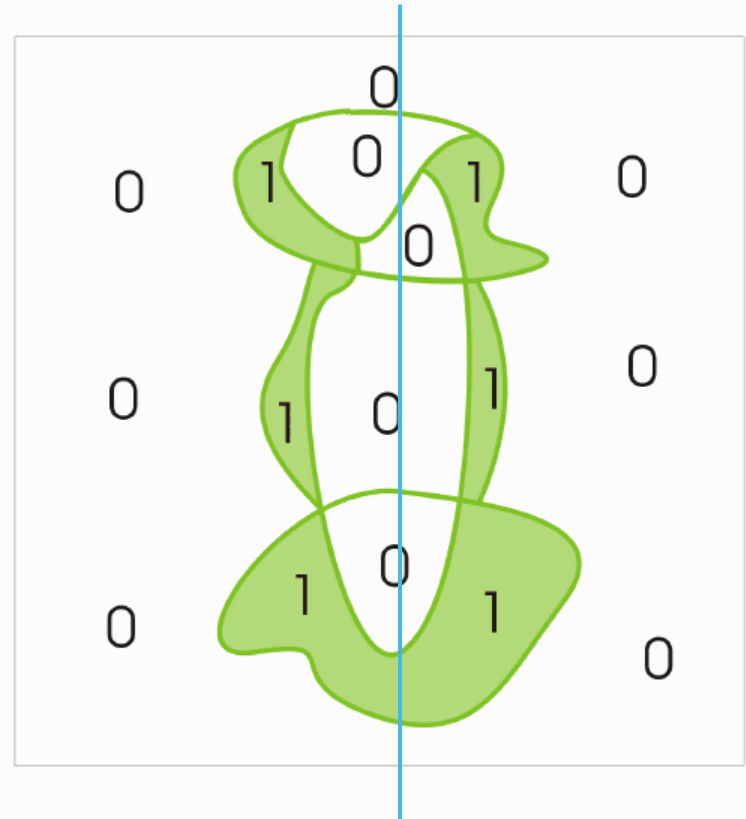
Segundo passo:

Depth Buffer



→  
Front to Back

Stencil Buffer



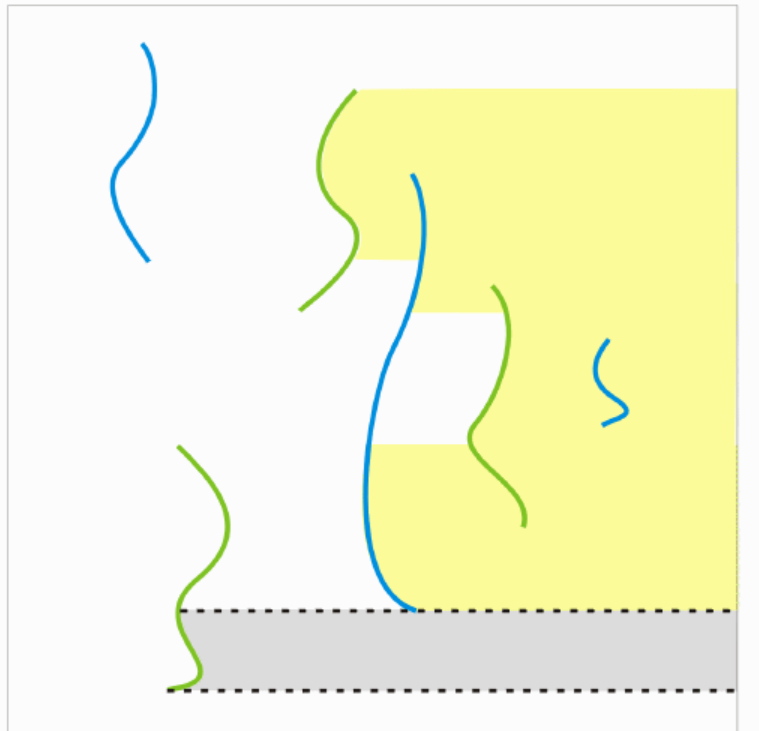
# Oclusão Implícita

## Algoritmo

Segundo passo:

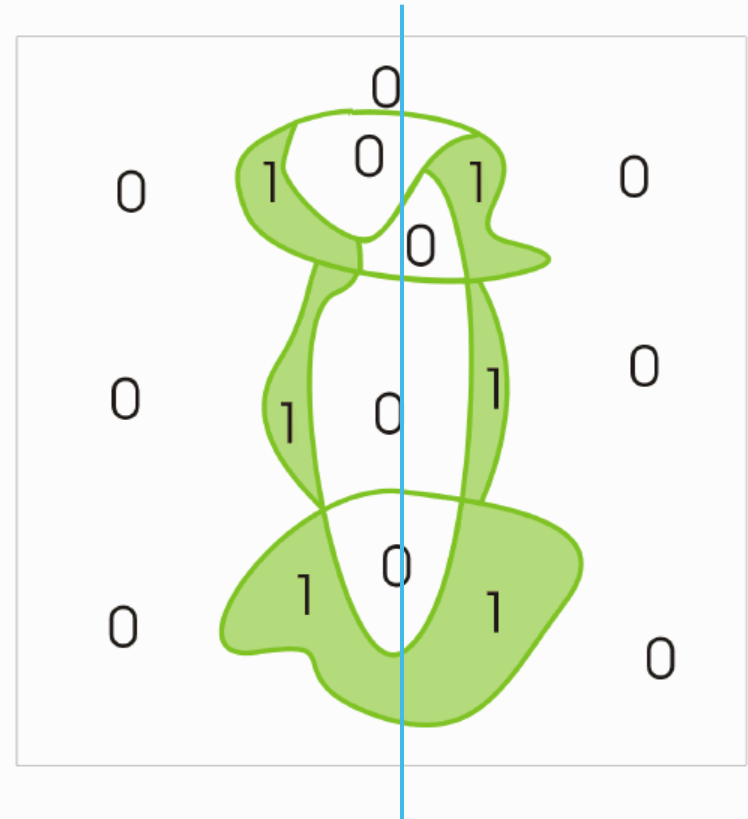
- A profundidade do occluder que está incompleto após o fim do segundo passo é marcado para infinito.

Depth Buffer



→  
Front to Back

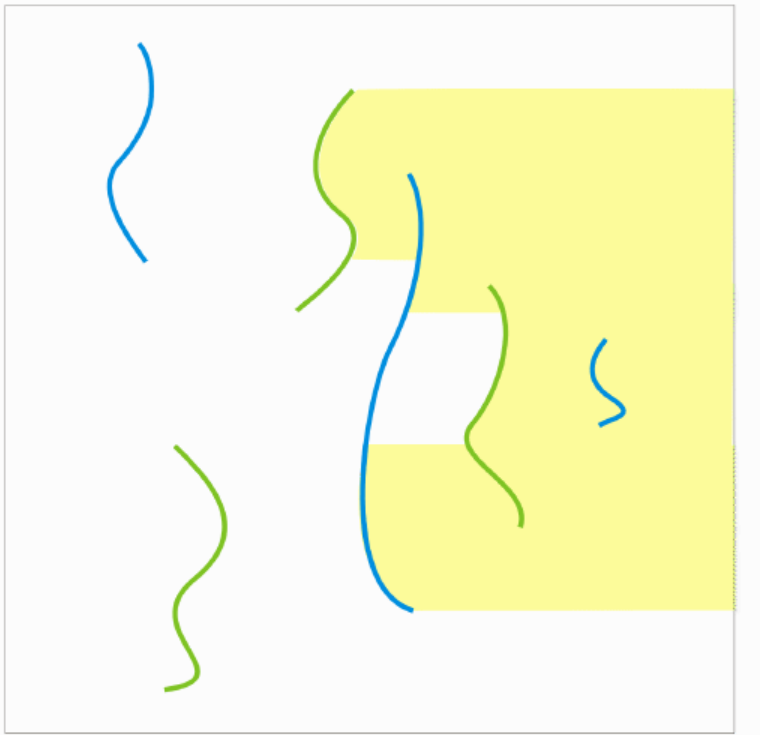
Stencil Buffer



# Oclusão Implícita

## Algoritmo

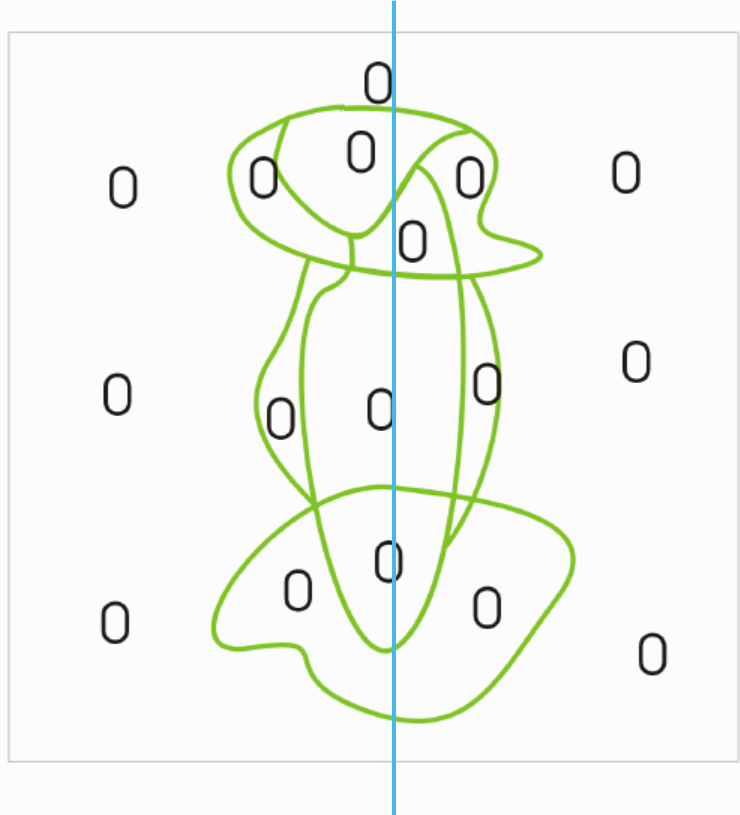
Depth Buffer



→  
Front to Back

∞

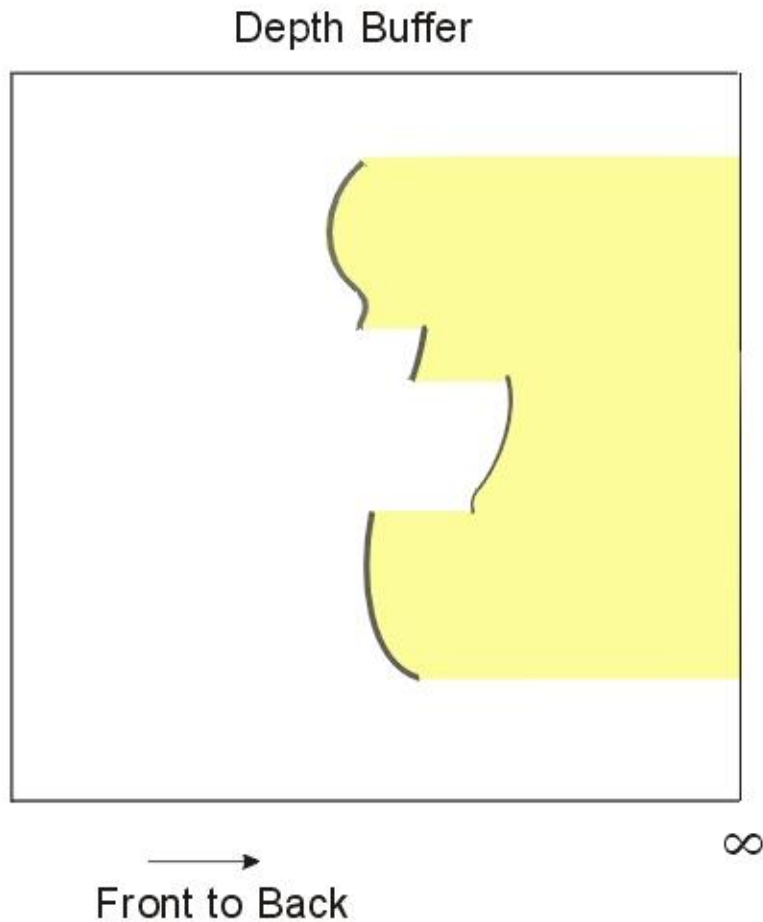
Stencil Buffer



# Oclusão Implícita

## Resultado

- A região em amarelo corresponde ao Ocluder Implícito.

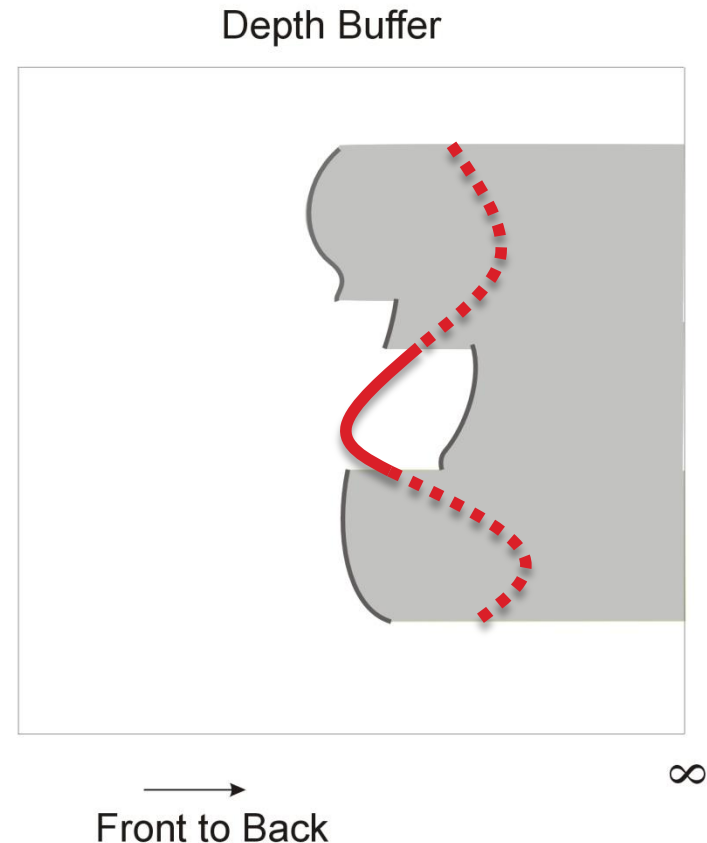




# Resultados

## Remoção de Linhas não-visíveis

- A oclusão implícita determina regiões que não são visíveis ao observador por estarem encobertas pela isosuperfície, assim as silhuetas contidas nessas regiões não são visíveis.

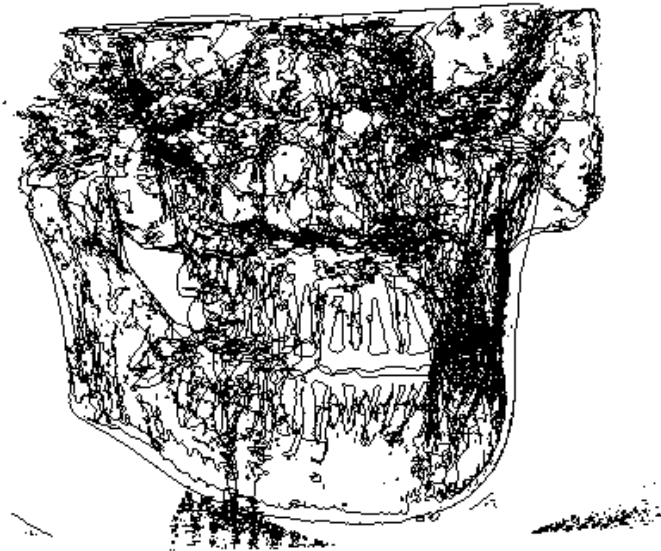




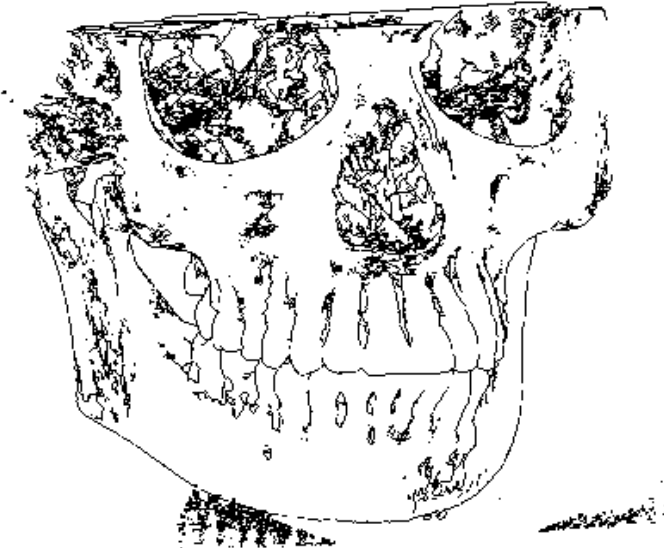
# Resultados

## Skull (512 x 512 x 512)

- Oclusão com octree profundidade 7



Sem Ocluder  
0,141 seg



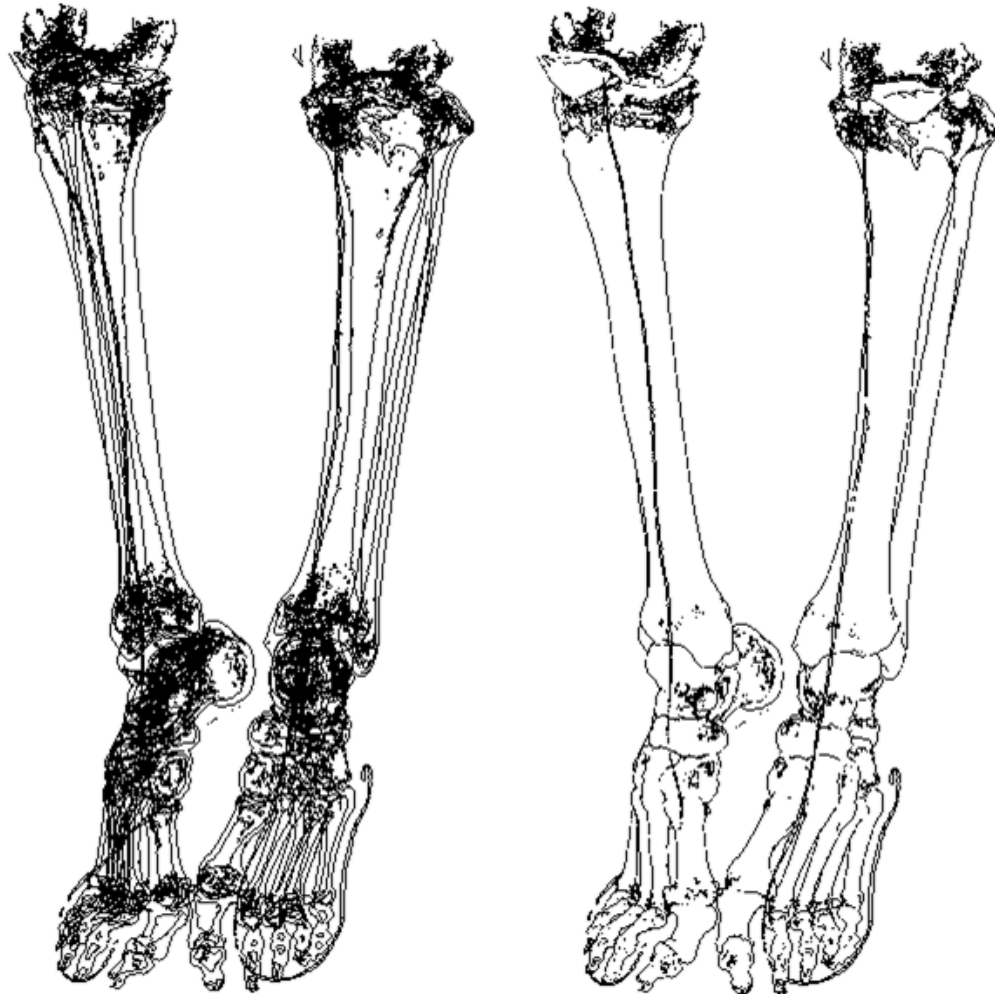
Com Ocluder  
0,151 seg



# Resultados

## Visible Woman (512 x 512 x 512)

Oclusão com octree  
profundidade 7



Sem Ocluder  
0,128 seg

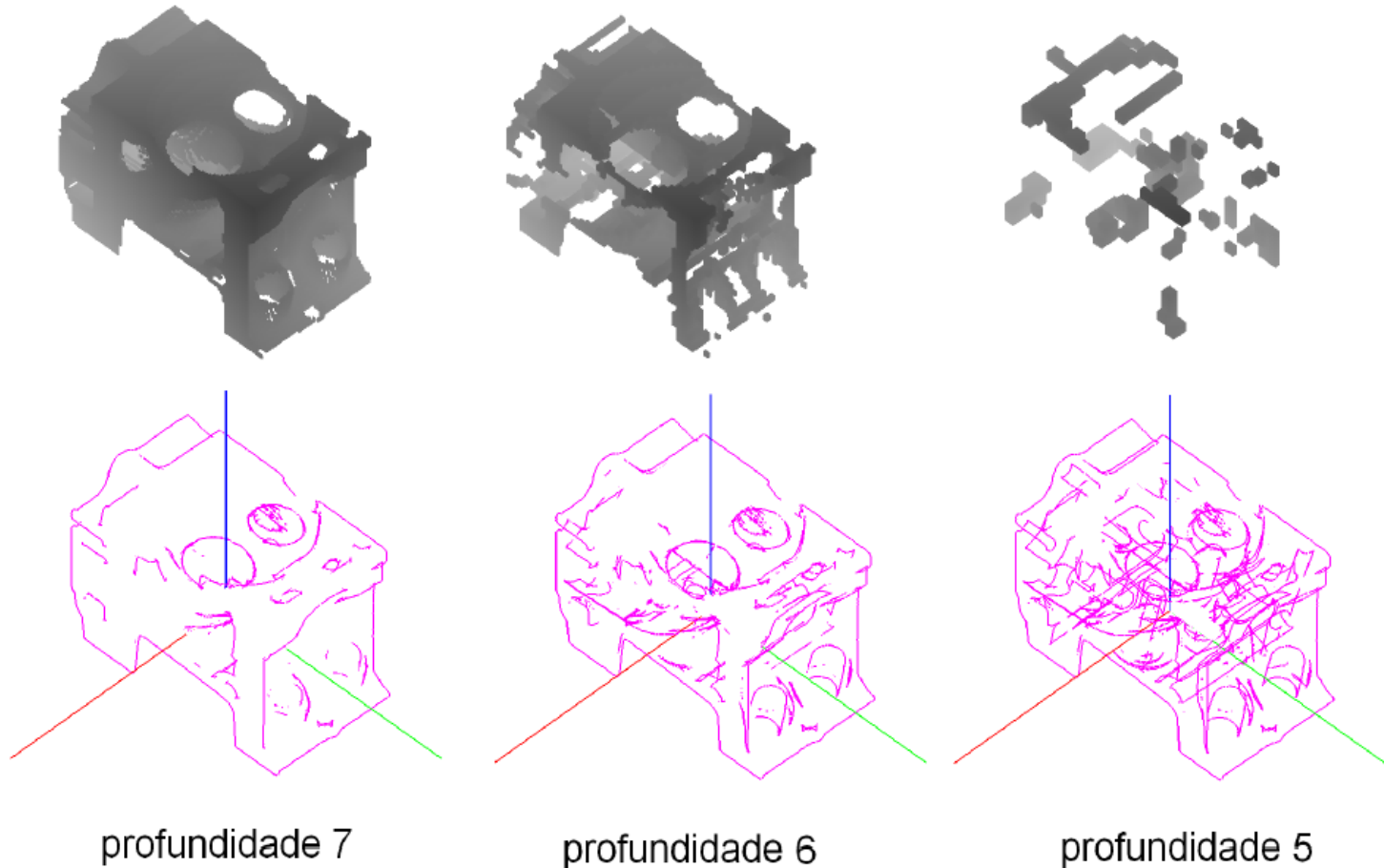
Com Ocluder  
0,137 seg



# Resultados

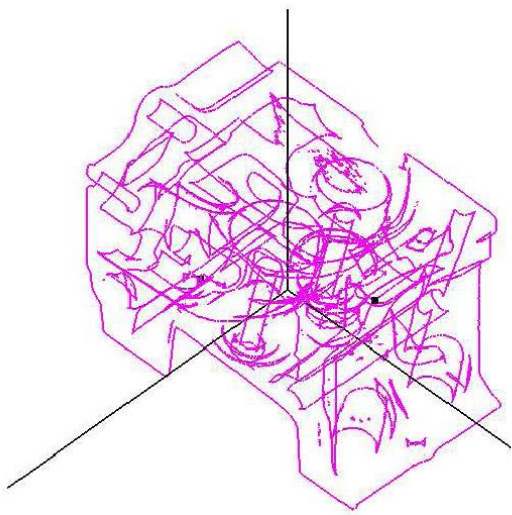
## Engine (256 x 256 x 128)

- Resultados com diferentes regiões de oclusão

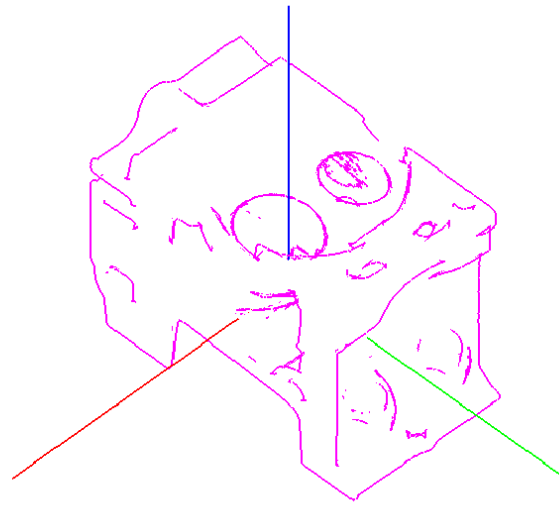


# Resultados

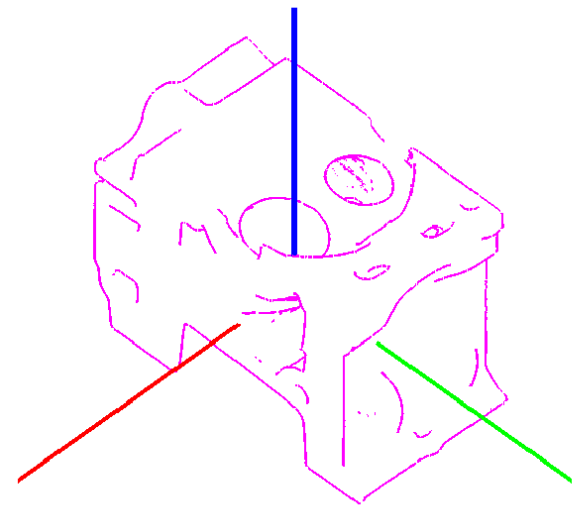
Engine (256 x 256 x 128)



Sem oclusão



Oclusão (prof. 7)



Silhueta com linhas  
não-visíveis  
removidas

# Conclusão

- A remoção de silhuetas não-visíveis foi satisfatória para os exemplos testados.
- O tempo de geração do ocluder não comprometeu os resultados e em termos de custo/benefício parecem ser razoáveis pelo ganho obtido em termos da remoção parcial das linhas não-visíveis.
- A oclusão implícita é gerada sem o cálculo explícito da isosuperfície, e permite modificar o isovalor com baixo custo.

## Limitações

- Semelhante a outras técnicas baseadas em visibilidade, a qualidade do resultado depende do nível de complexidade da isosuperfície envolvida.
- A oclusão implícita é adequada para domínios convexos.