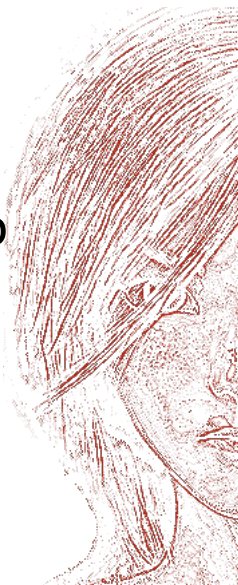


# Detecção de Colisão

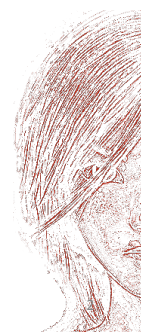
Márcio Sarroglia Pinho  
Isabel Harb Manssour



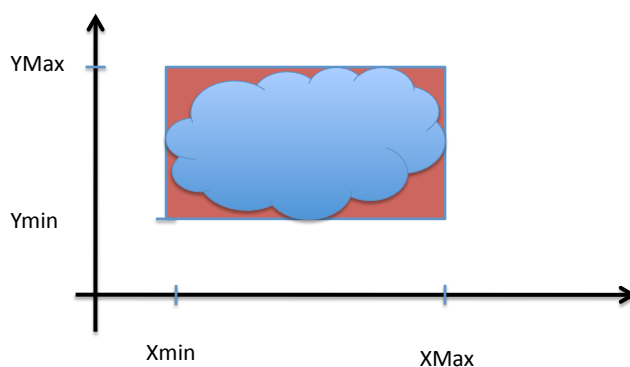
1

## ENVELOPES / BOUNDING BOXES

Computação Gráfica 2D



## AABB – Axis Aligned Bounding Boxes (Envelopes)



Computação Gráfica 2D

3

## Teste de Colisão entre AABB

```
typedef struct Ponto{  
    int x, y;  
} Ponto;
```

```
typedef struct AABB{  
    Ponto Centro;  
    Ponto MeiaLarg;  
} AABB;
```

Computação Gráfica 2D

4

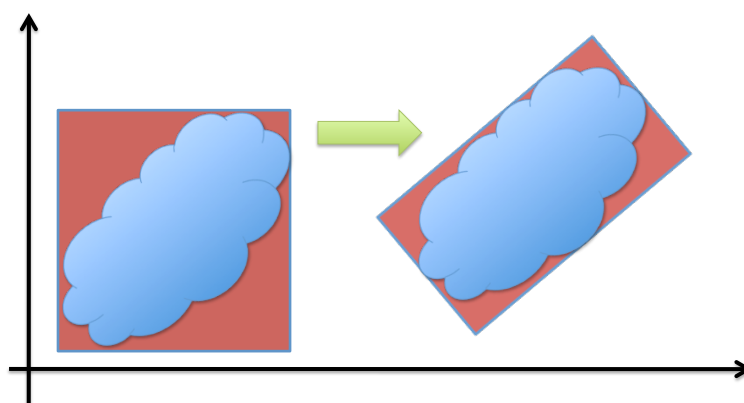
## Teste de Colisão entre AABB

```
bool testAABBAABB(const AABB &E1, const AABB &E2)
{
    if ( Abs(E1.Centro.x - E2.Centro.x) >
          (E1.MeiaLarg.x + E2.MeiaLarg.x))
        return false; // não há colisão
    // idem para Y e Z
    return true; // Há colisão
};
```

Computação Gráfica 2D

5

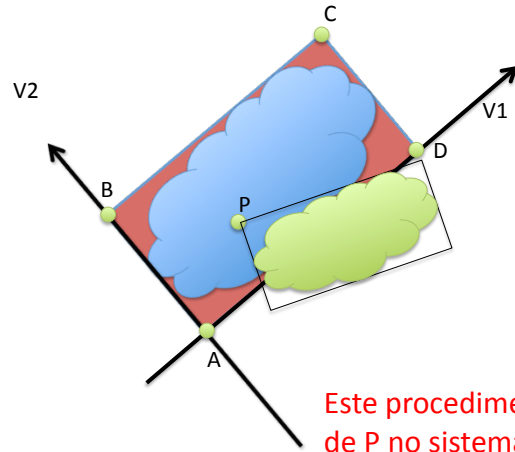
## OABB – Object Oriented Bounding Boxes



Computação Gráfica 2D

6

## Colisão entre OOBB



$$M = \begin{bmatrix} V1.X & V1.Y \\ V2.X & X2.Y \end{bmatrix}$$

V1 e V2: vetores unitários

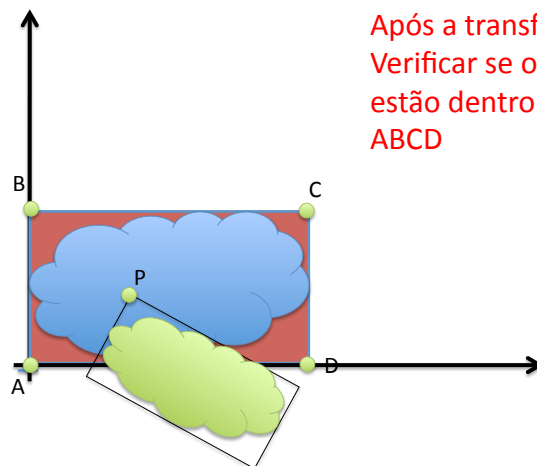
$$M \times P'' = I \times P$$

$$P'' = P \times M^{-1}$$

$$M^{-1} = M^T$$

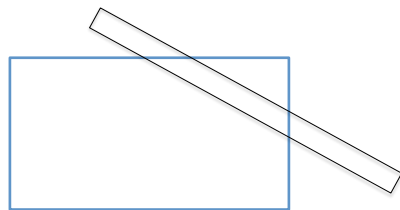
Este procedimento obtém as coordenadas de P no sistema e coordenadas V1-V2

## Colisão entre OOBB

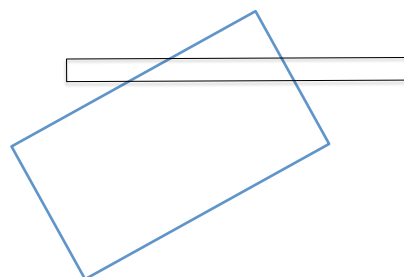


Após a transformação...  
Verificar se os vértices P estão dentro do envelope ABCD

## Colisão entre OOBB



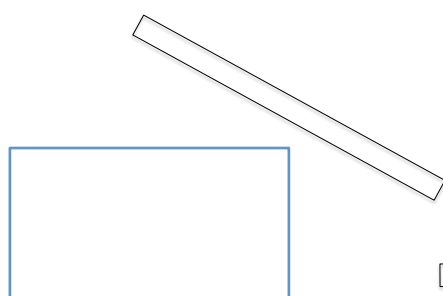
Em casos como estes será necessário verificar se há SEPARAÇÃO ENTRE EIXOS



Computação Gráfica 2D

9

## Colisão entre OOBB



Em casos como estes será necessário verificar se há SEPARAÇÃO ENTRE EIXOS



Computação Gráfica 2D

10

## Envelopes Hierárquicos

- Subdivisão em Partes

- Manual

- Depende do conhecimento do objeto

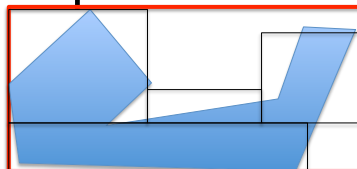
- Automática

- Subdivisão Regular

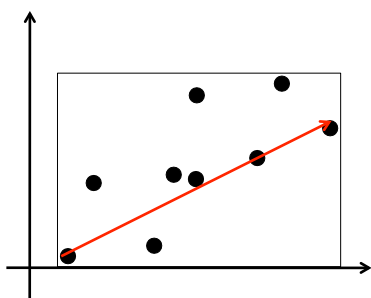
- Quadriculado

- Adaptativa

- Quadrees



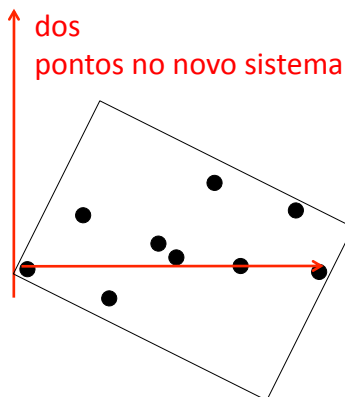
## Obtenção de uma OOBB



- Obter os dois pontos mais distantes, criando o eixo principal

- Criar um novo sistema de coordenadas a partir do maior eixo

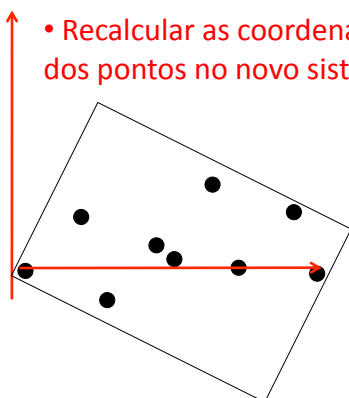
- Recalcular as coordenadas dos pontos no novo sistema



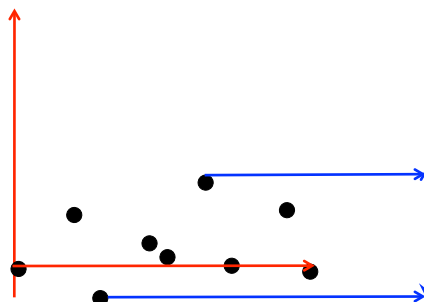
## Obtenção de uma OOB

- Criar um novo sistema de coordenadas a partir do maior eixo

- Recalcular as coordenadas dos pontos no novo sistema

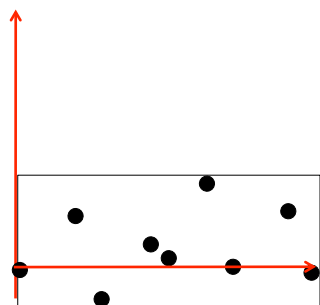


- Encontrar a maior distância vertical entre os pontos

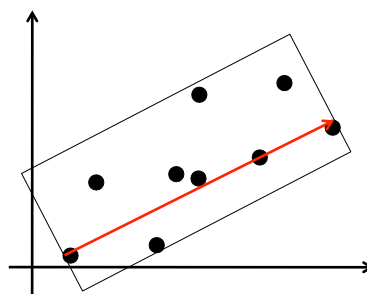


## Obtenção de uma OOB

- Construir uma AABB

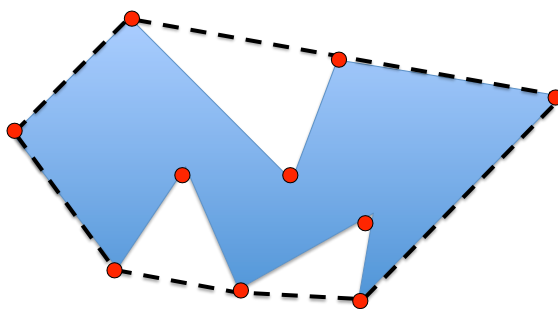


- Retornar ao sistema de coordenadas original

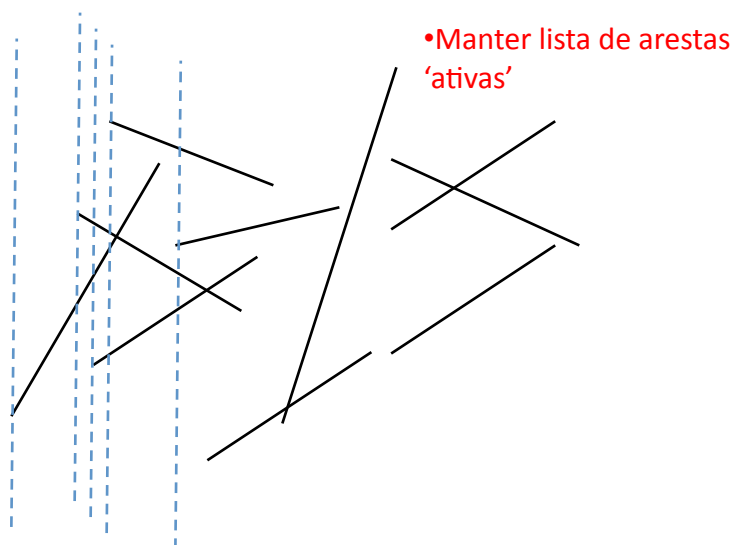


## Cobertura Convexa

- Convex Hull
  - Quick hull
  - Half space



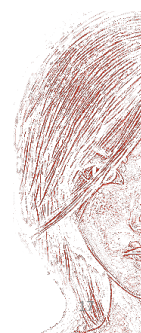
## Algoritmo de Plane-Sweep





## SUBDIVISÃO DO ESPAÇO

Computação Gráfica 2D

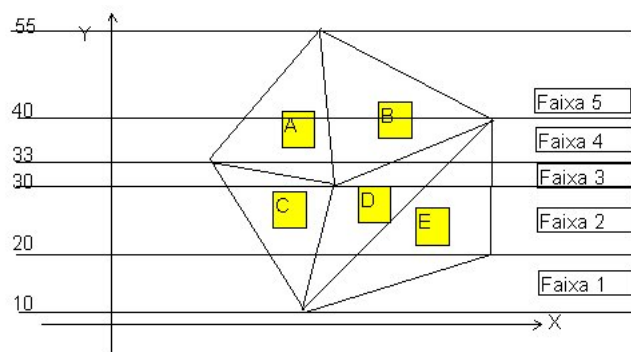


## Subdivisão do Espaço

- Portal Culling
- Slabs - Faixas
- Diagrama de Voronoi
- Triangulação de Delaunay
- Binary Search Partition

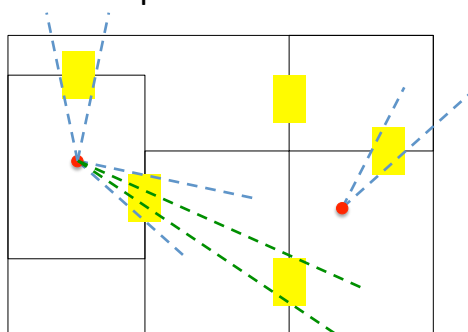
## Subdivisão do Espaço

- Slabs - Faixas



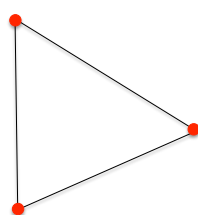
## Subdivisão do Espaço

- Portal Culling
  - Útil para ambientes onde há 'portas'



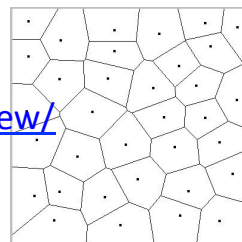
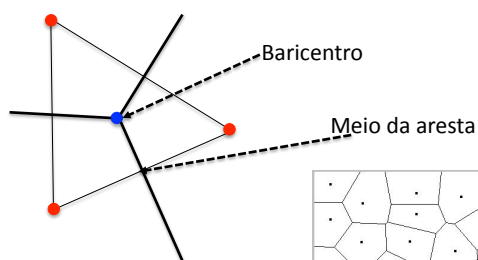
## Subdivisão do Espaço

- Diagrama de Voronoi
  - Determina regiões mais próximas de um ponto



– Applet:

<http://www.cs.cornell.edu/home/chew/Delaunay.html>



## Subdivisão do Espaço

- Triangulação de Delaunay
- Applet:
  - <http://www.cs.cornell.edu/home/chew/Delaunay.html>

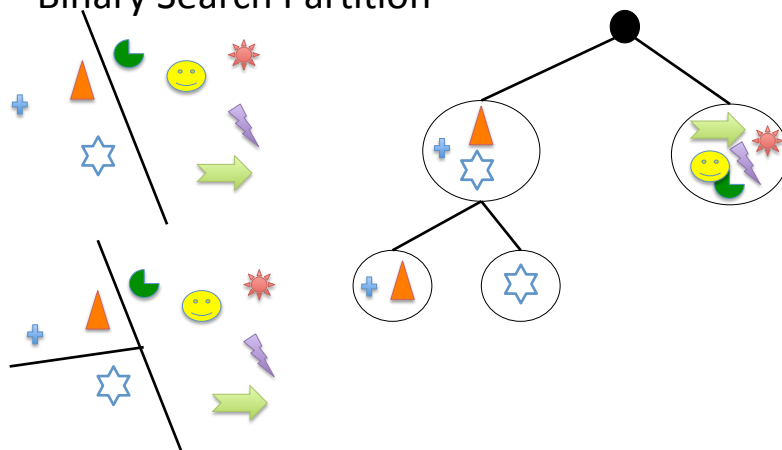
## Subdivisão do Espaço

- Binary Search Partition
  - Subdivide o espaço em semi-planos
  - Testa colisão apenas com objetos que estão no mesmo semi-plano



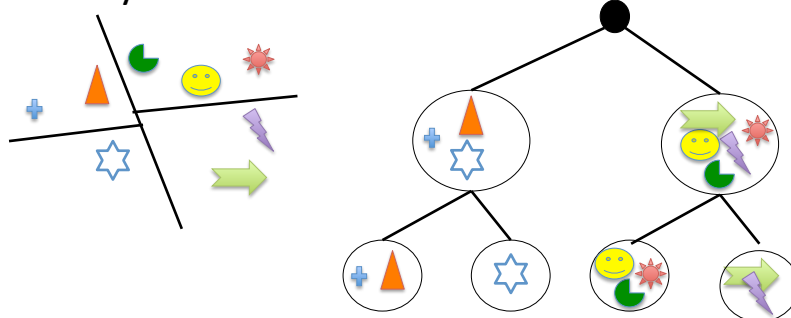
## Subdivisão do Espaço

- Binary Search Partition



## Subdivisão do Espaço

- Binary Search Partition



## Subdivisão do Espaço

- Binary Search Partition
  - Utilizado no DOOM para remoção de elementos ocultos
  - Ocupa apenas a memória já ocupada pelas faces do objeto
  - Applet: <http://pauillac.inria.fr/~levy/bsp/>