

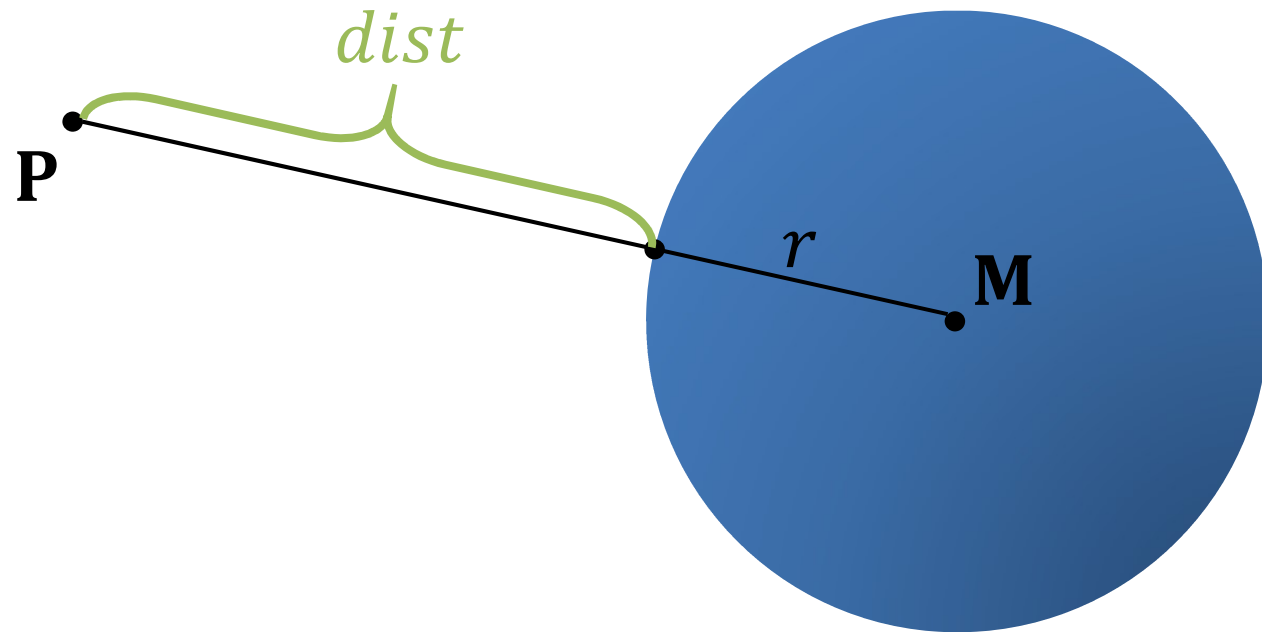
# Raymarching Distance Fields

# Distance Functions

- Distance function sphere

$$\|\mathbf{P} - \mathbf{M}\| - r = \textit{dist}_{\textit{signed}}(\mathbf{P}) = \textit{dist}_s(\mathbf{P})$$

$$\max(0, \|\mathbf{P} - \mathbf{M}\| - r) = \textit{dist}_{\textit{unsigned}}(\mathbf{P}) = \textit{dist}_u(\mathbf{P})$$

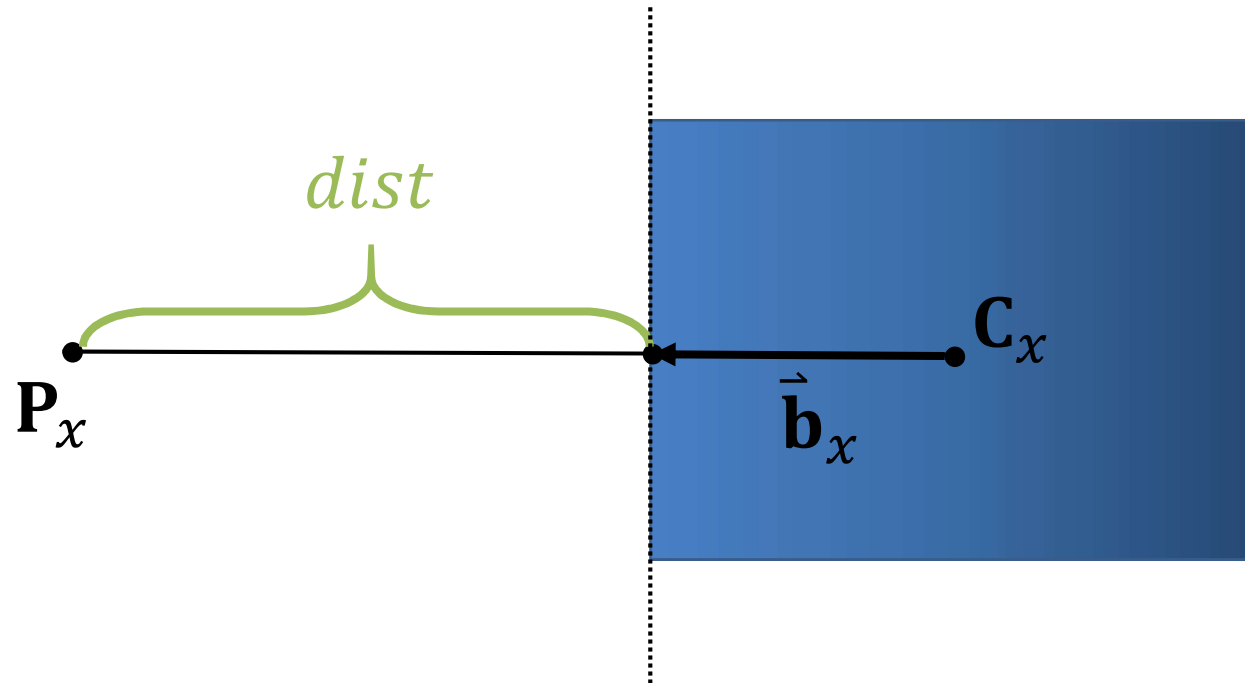


# Distance Functions

- Unsigned distance function box –  $x$ -direction

$$\text{dist}_u(\mathbf{P}_x) = \max(\text{abs}(\mathbf{P}_x - \mathbf{C}_x) - \vec{\mathbf{b}}_x, 0)$$

where  $\text{abs}(\vec{\mathbf{x}})$  is the component-wise absolute value of  $\vec{\mathbf{x}}$

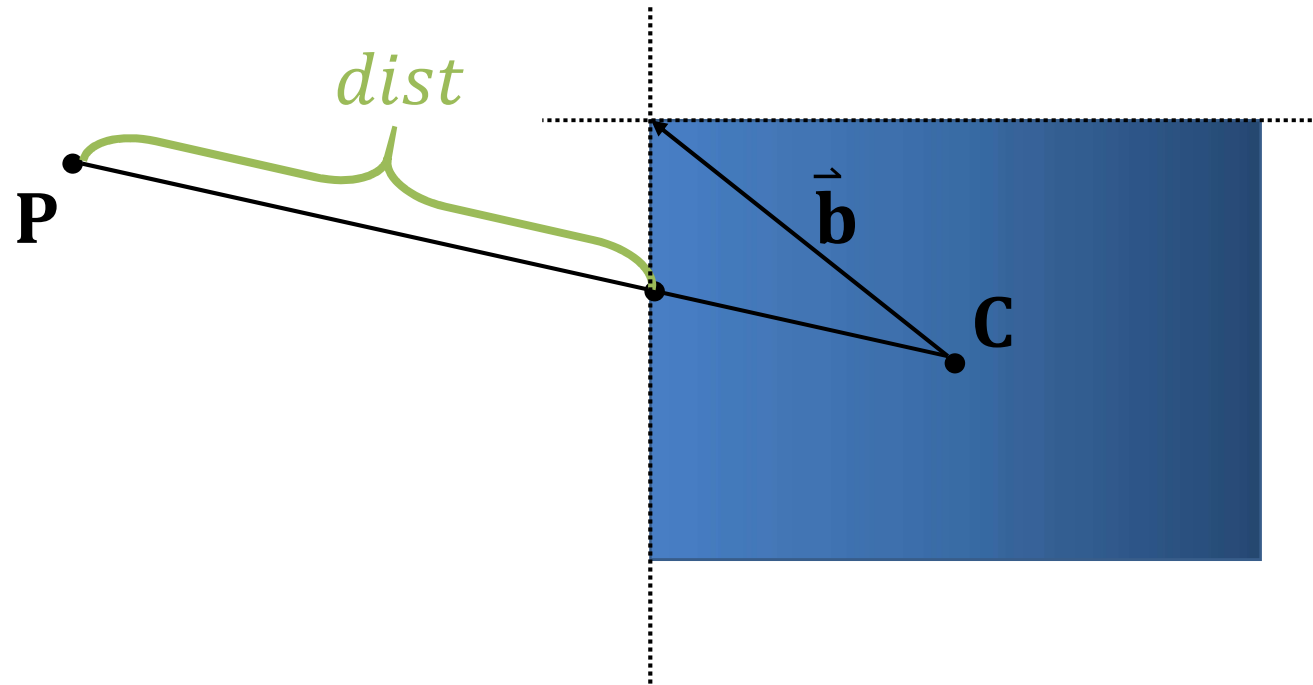


# Distance Functions

- Unsigned distance function box

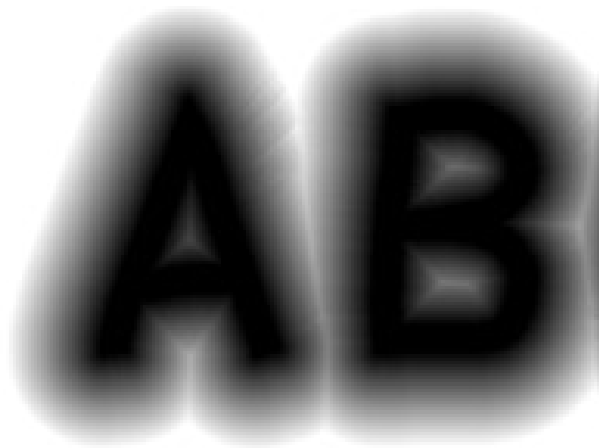
$$\text{dist}_u(\mathbf{P}) = \|\max(\text{abs}(\mathbf{P} - \mathbf{C}) - \vec{\mathbf{b}}, \vec{\mathbf{0}})\|$$

where  $\|\vec{\mathbf{x}}\|$  is the vector absolute value of  $\vec{\mathbf{x}}$

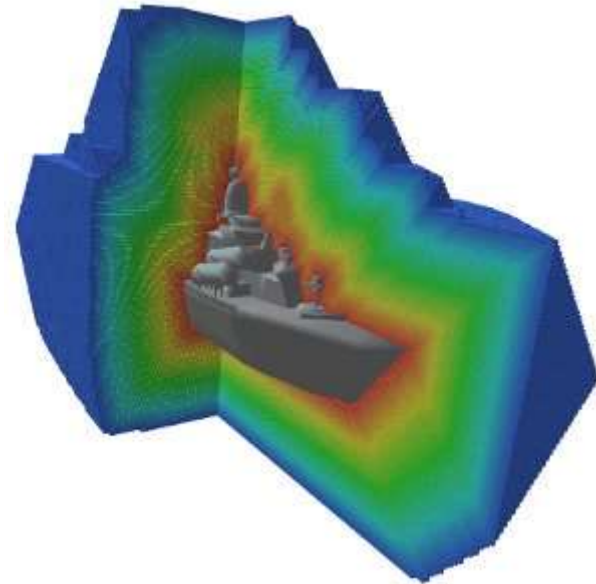


# Distance Fields

$$\mathbb{R}^2 \rightarrow \text{dist}(\mathbb{R}^2)$$

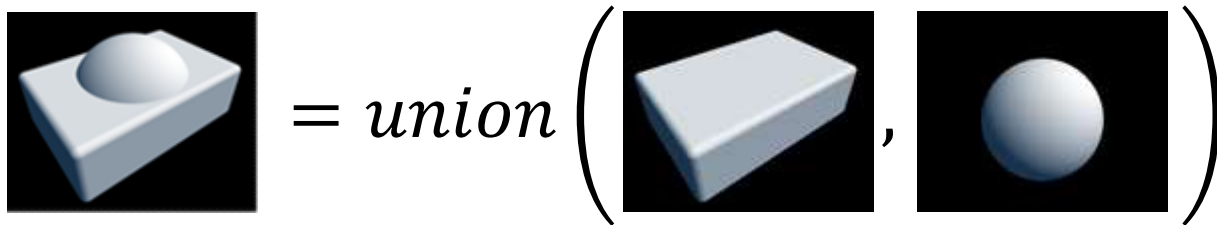


$$\mathbb{R}^3 \rightarrow \text{dist}(\mathbb{R}^3)$$



# Operations on Distance Fields

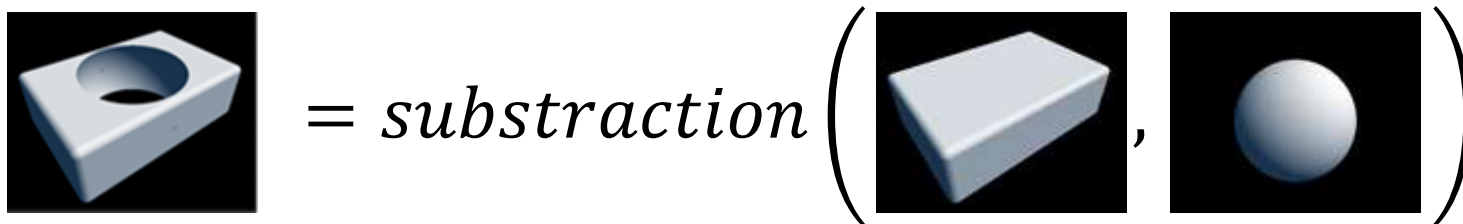
- Given  $dist_1(\mathbb{R}^3)$  and  $dist_2(\mathbb{R}^3)$



The image shows the union of a rectangular box and a sphere. On the left is a 3D rendering of a light blue box with a sphere on top. To its right is an equals sign followed by the word "union" in italics, then a large left parenthesis. Inside the parenthesis are two separate 3D renderings: a light blue box and a light blue sphere. This is followed by a comma and another light blue sphere, and finally a large right parenthesis.

$$\text{Union} = \text{union} \left( \text{Box}, \text{Sphere} \right)$$

- The union is  $\min(dist_1(\mathbb{R}^3), dist_2(\mathbb{R}^3))$



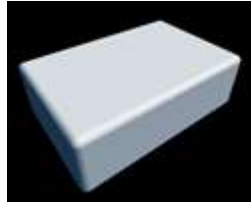
The image shows the subtraction of a sphere from a box. On the left is a 3D rendering of a light blue box with a circular hole in the center. To its right is an equals sign followed by the word "subtraction" in italics, then a large left parenthesis. Inside the parenthesis are two separate 3D renderings: a light blue box and a light blue sphere. This is followed by a comma and another light blue sphere, and finally a large right parenthesis.

$$\text{Subtraction} = \text{subtraction} \left( \text{Box}, \text{Sphere} \right)$$

- The subtraction is  $\max(-dist_1(\mathbb{R}^3), dist_2(\mathbb{R}^3))$

# Operations on Distance Fields

- Given  $dist(\mathbb{R}^3) =$



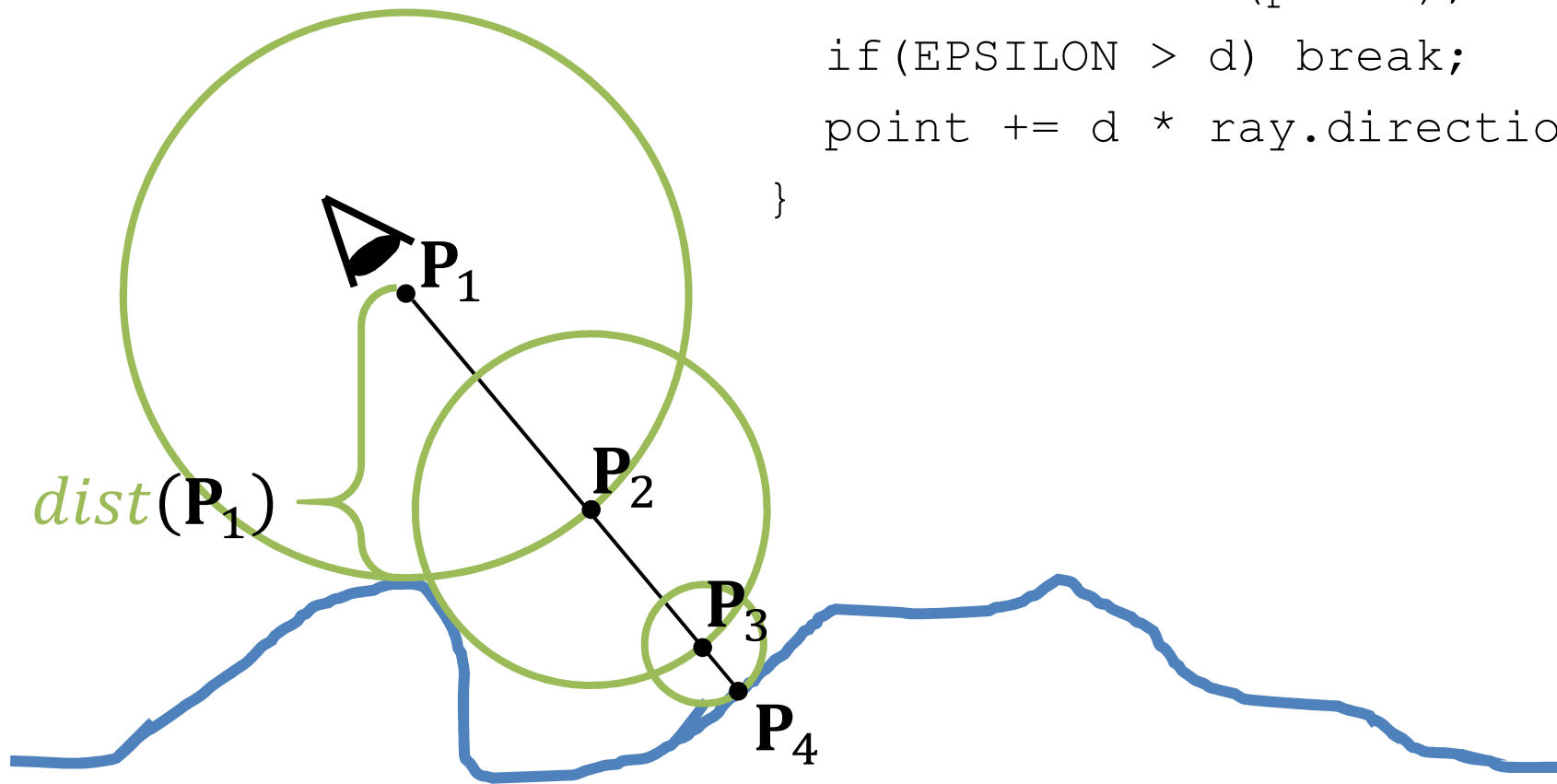
$$= dist(repeat(\mathbb{R}^3))$$

- Repeat is  $\text{mod}(\mathbf{P}, \vec{\mathbf{b}}) - \frac{1}{2} \vec{\mathbf{b}}$   
where  $\text{mod}(\vec{\mathbf{a}}, \vec{\mathbf{c}})$  is component-wise  $\vec{\mathbf{a}}$  modulo  $\vec{\mathbf{c}}$

# Raymarching Distance Fields

- *dist*( $P_i$ )

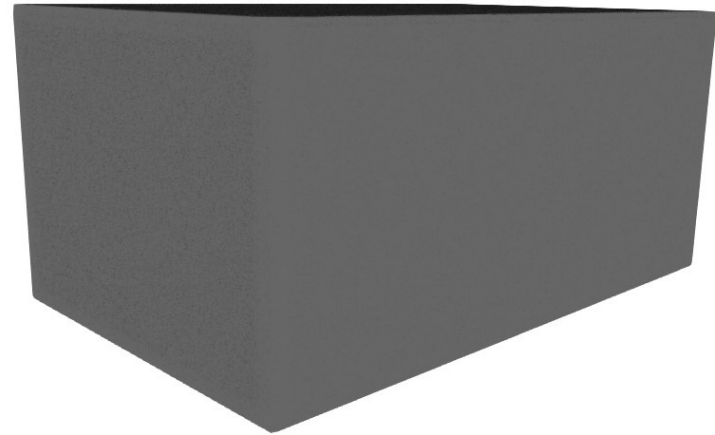
```
vec3 point = ray.origin;  
while(--maxSteps) {  
    float d = dist(point);  
    if(EPSILON > d) break;  
    point += d * ray.direction;  
}
```





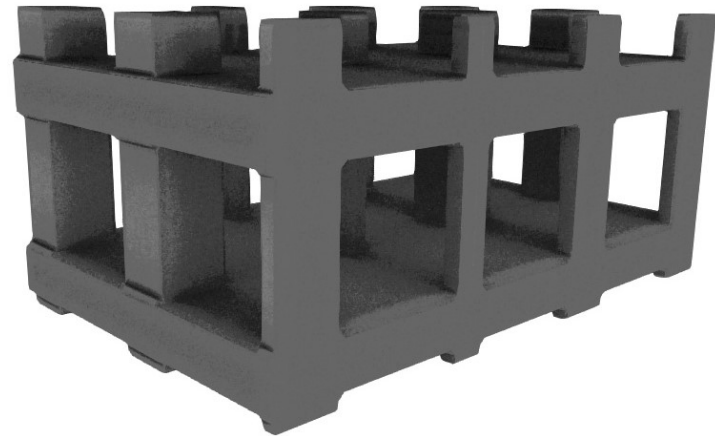
# A Box

```
Box(pos, size)
{
    a = abs(pos-size) - size;
    return max(a.x, a.y, a.z);
}
```



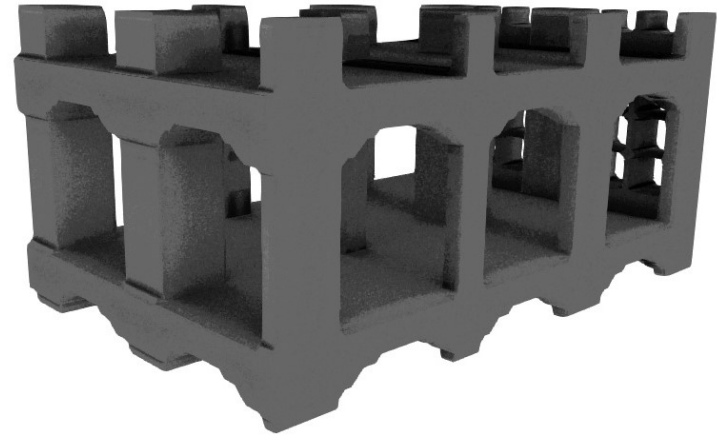
# Cutting with Booleans

```
d = Box(pos)
c = fmod(pos * A, B)
subD = max(c.y, min(c.y, c.z))
d = max(d, -subD)
```



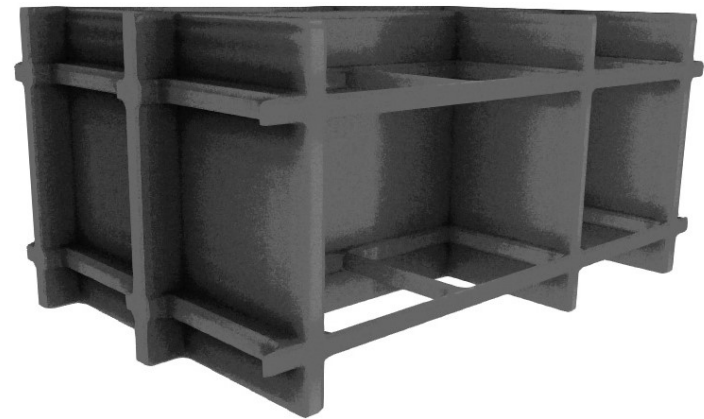
# More Booleans

```
d = Box(pos)
c = fmod(pos * A, B)
subD = max(c.y, min(c.y, c.z))
subD = min(subD, cylinder(c))
subD = max(subD, Windows())
d = max(d, -subD)
```



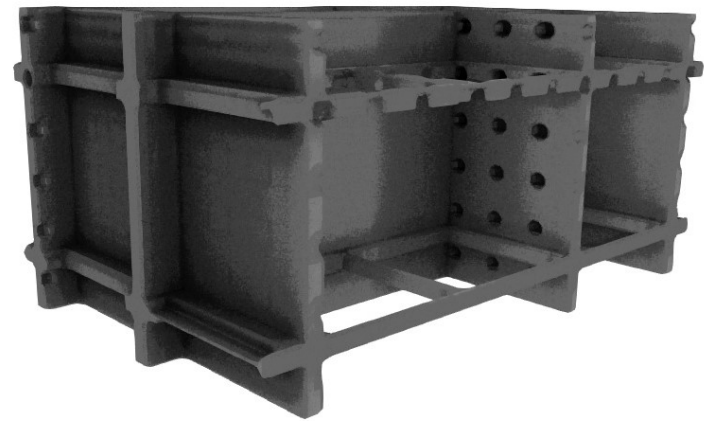
# Repeated Booleans

```
d = Box(pos)
e = fmod(pos + N, M)
floorD = Box(e)
d = max(d, -floorD)
```



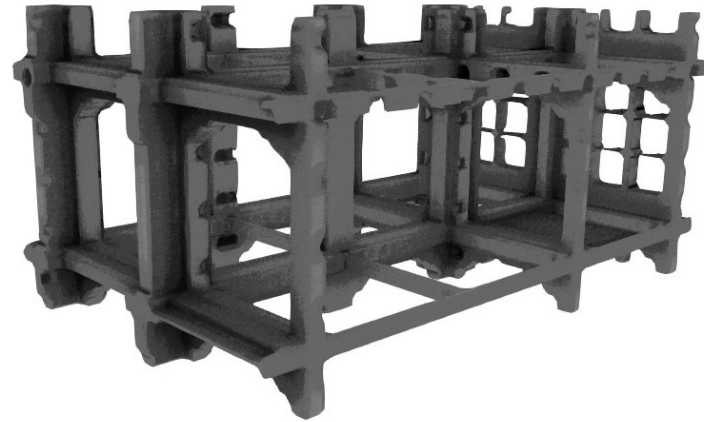
# Cutting Holes

```
d = Box(pos)
e = fmod(pos + N, M)
floorD = Box(e)
floorD = min(floorD, holes())
d = max(d, -floorD)
```



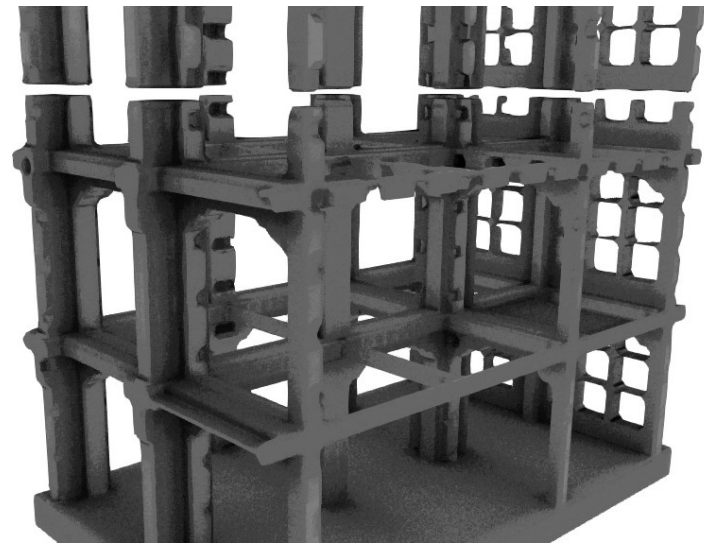
# Combined Result

```
d = Box(pos)
c = fmod(pos * A, B)
subD = max(c.y, min(c.y, c.z))
subD = min(subD, cylinder(c))
subD = max(subD, Windows())
e = fmod(pos + N, M)
floorD = Box(e)
floorD = min(floorD, holes())
d = max(d, -subD)
d = max(d, -floorD)
```



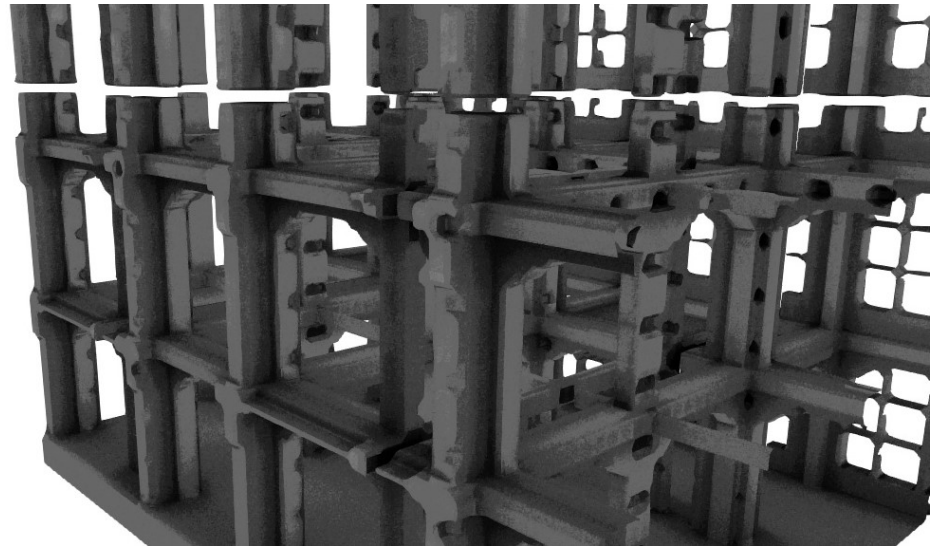
# Repeating the Space

```
pos.y = frac(pos.y)
d = Box(pos)
c = fmod(pos * A, B)
subD = max(c.y,min(c.y,c.z))
subD = min(subD,cylinder(c))
subD = max(subD, Windows())
e = fmod(pos + N, M)
floorD = Box(e)
floorD = min(floorD,holes())
d = max(d, -subD)
d = max(d, -floorD)
```



# Repeating the Space

```
pos.xy = frac(pos.xy)
d = Box(pos)
c = fmod(pos * A, B)
subD = max(c.y,min(c.y,c.z))
subD = min(subD,cylinder(c))
subD = max(subD, Windows())
e = fmod(pos + N, M)
floorD = Box(e)
floorD = min(floorD,holes())
d = max(d, -subD)
d = max(d, -floorD)
```

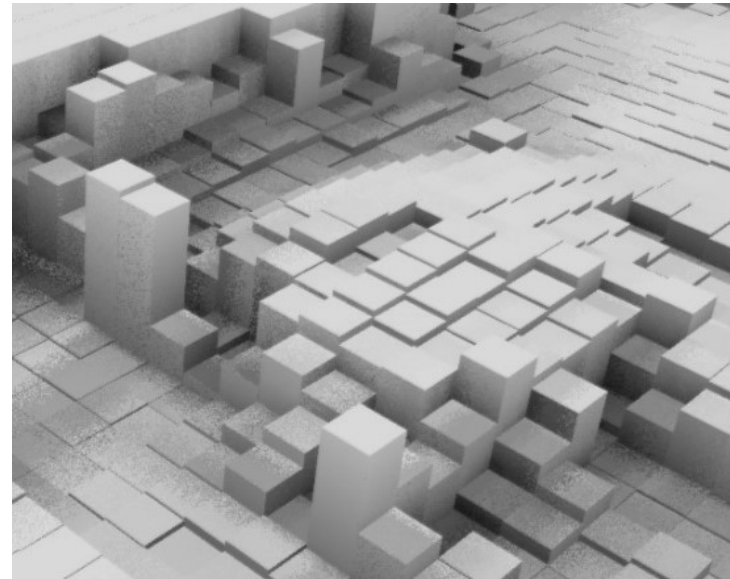




# Height Field

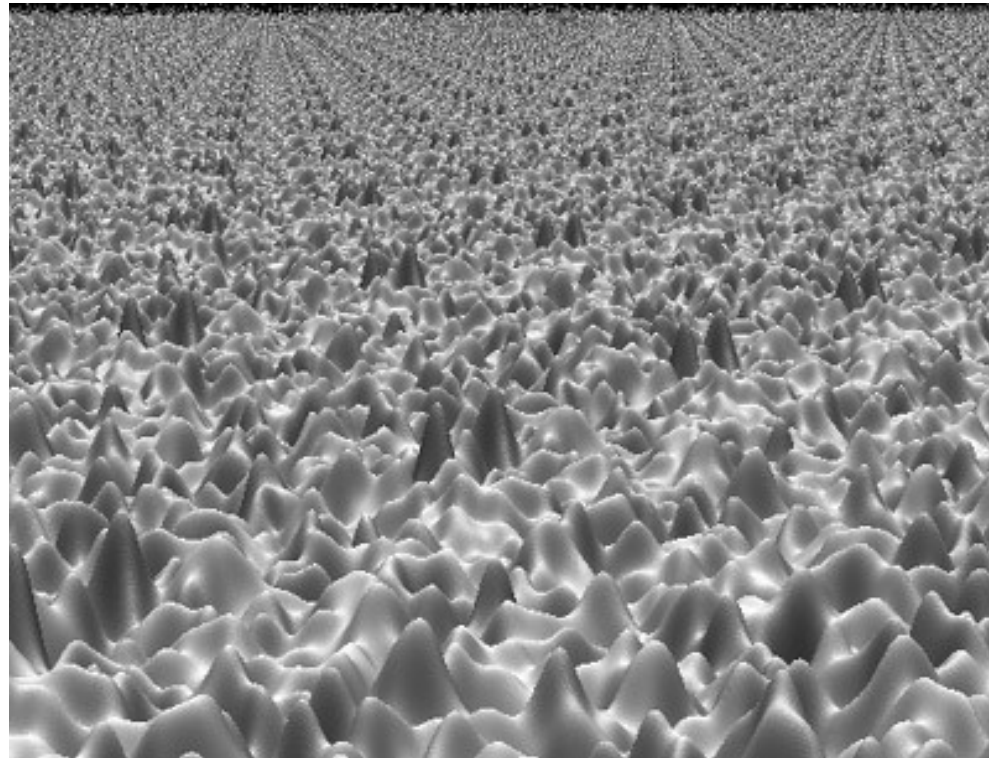
- Grid that stores a height at each position

1	2	1	2	2	1
2	1	2	1	10	1
3	1	2	6	9	0
4	1	2	5	0	0
5	1	2	3	0	0
6	1	2	2	1	0



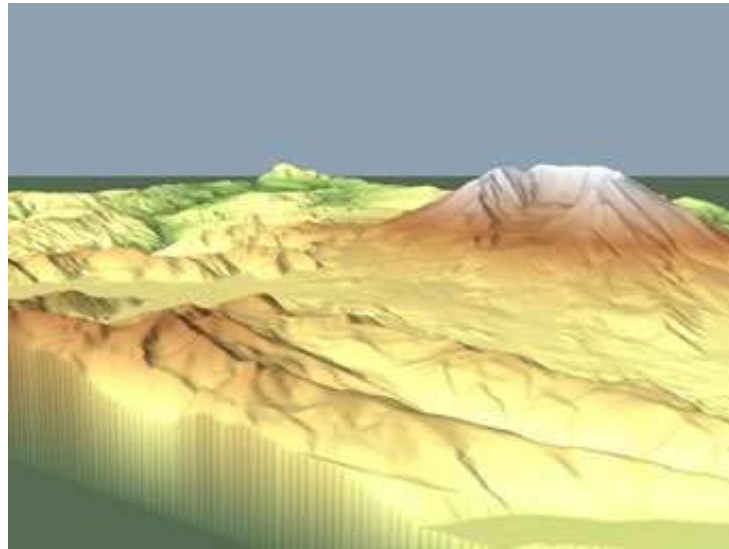
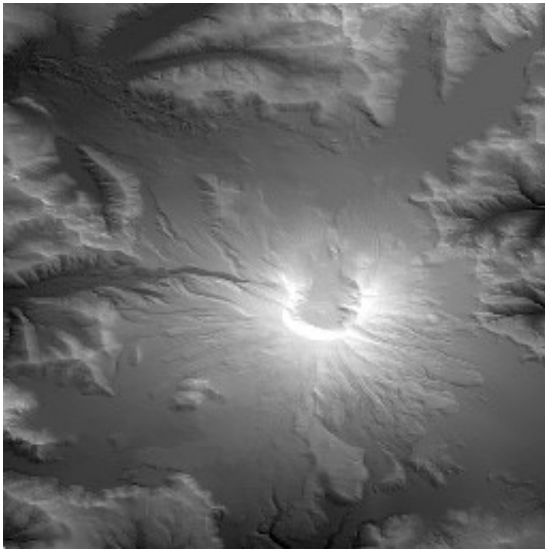
# Height Field

- Can use mathematical function to create grid values



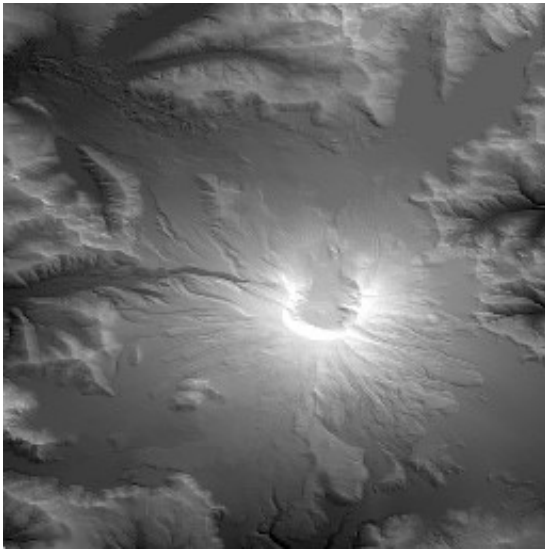
# Height Field

- Can use texture to store grid



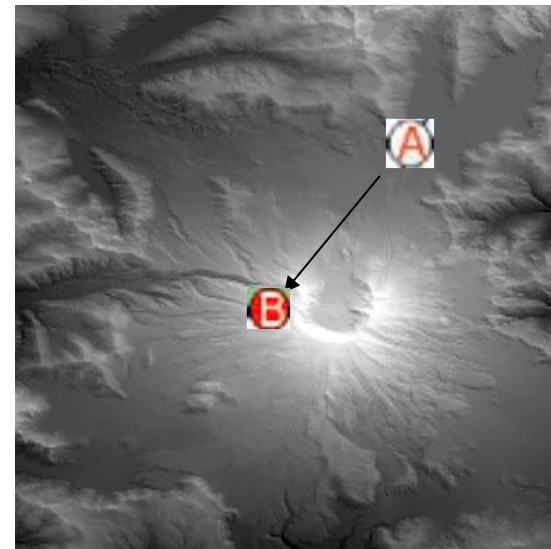
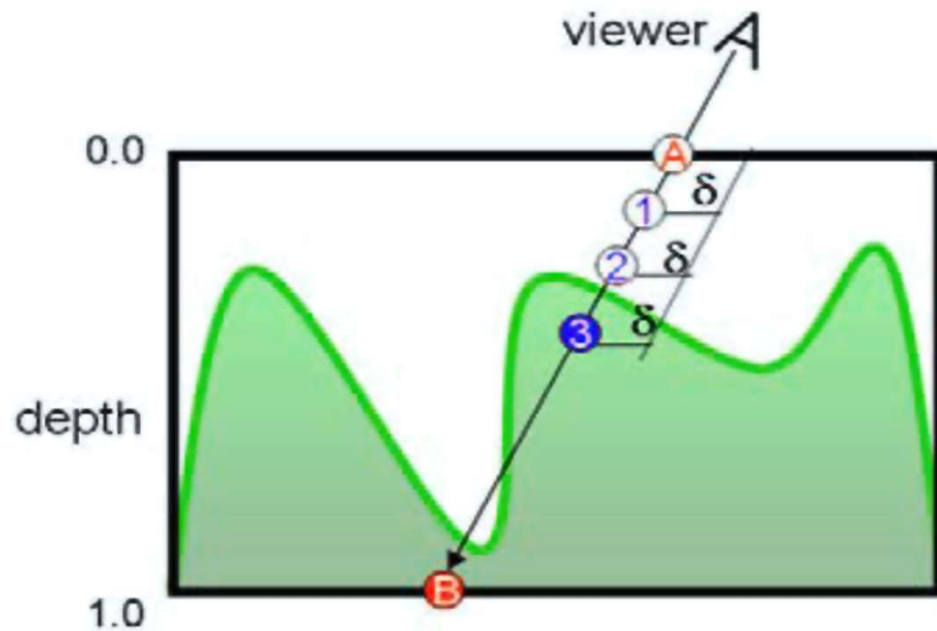
# Height Field

- Can render with different methods



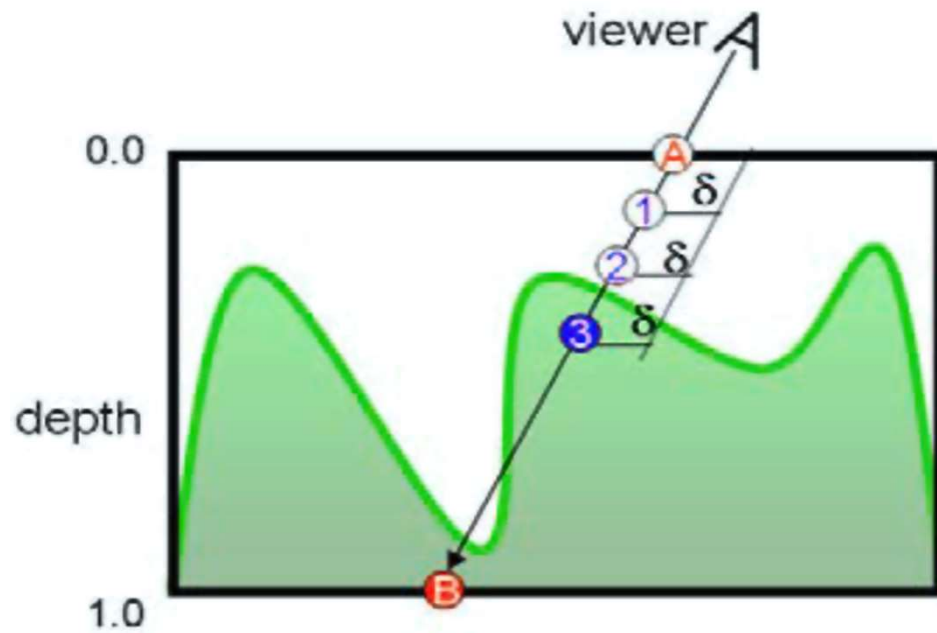
# Raymarching Height Field

- Step with small increments along ray
- Interval bisection

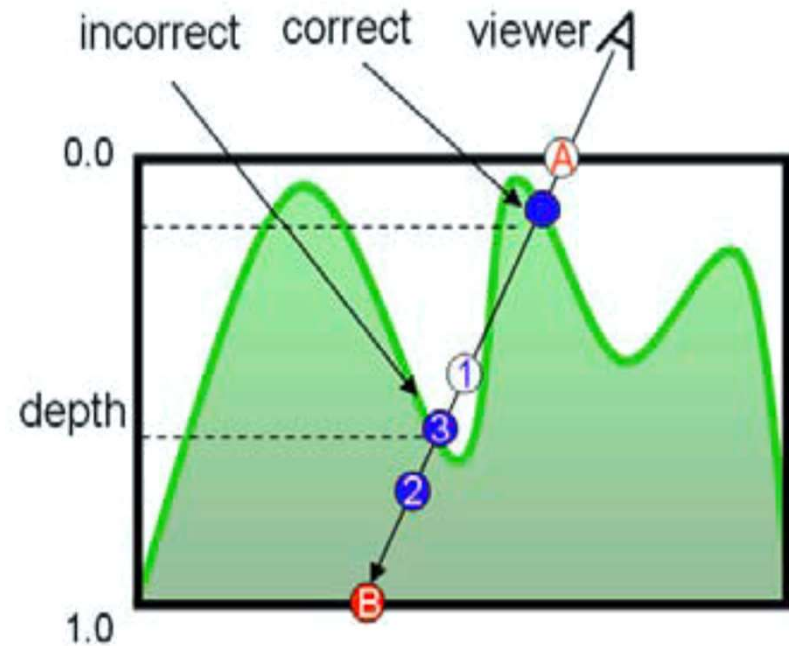


# Raymarching Height Field

- Step with small increments along ray

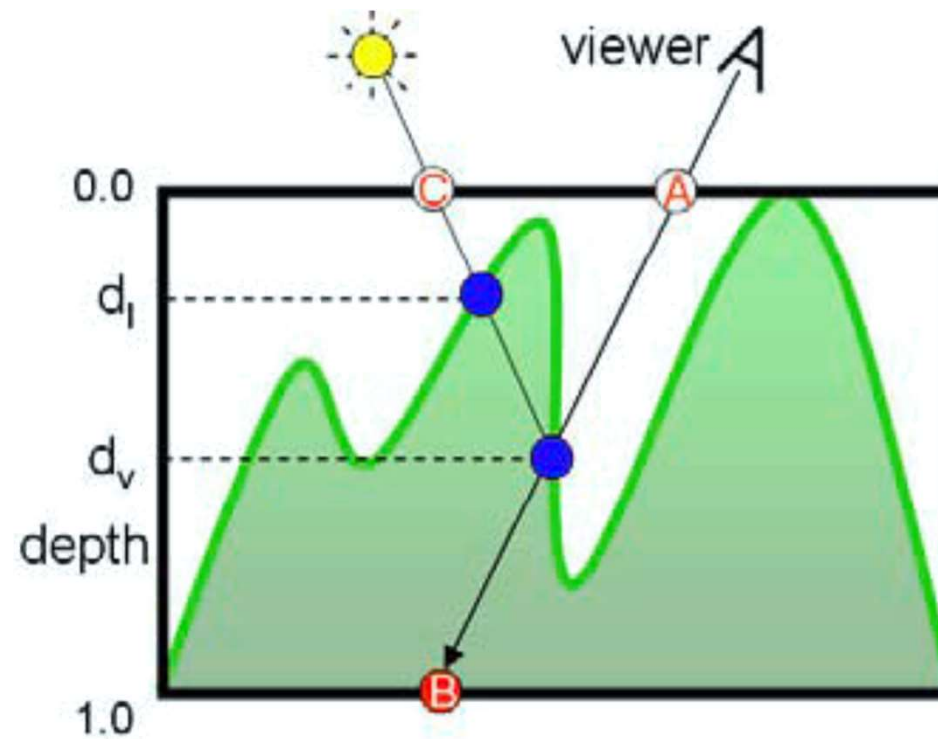


- Interval bisection



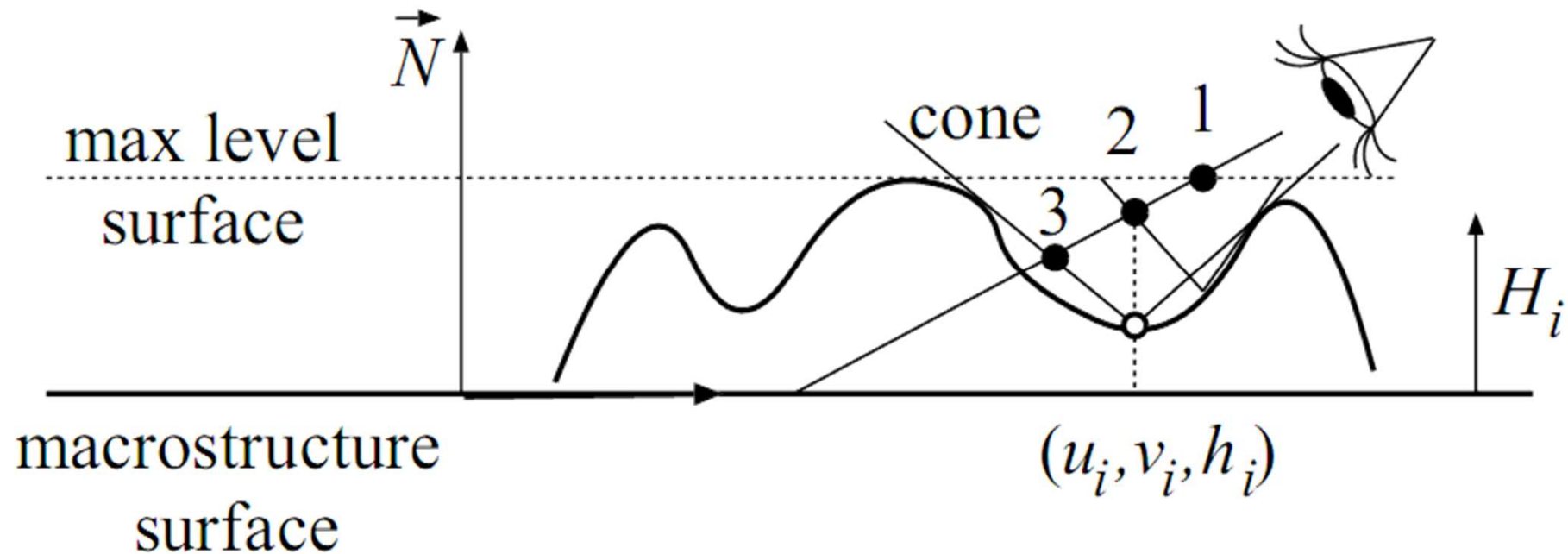
# Shadowing

- Send shadow feeler ray



# Accelerating Heightfield Rendering

- Help texture
  - Each texel stores cone of empty space above
  - Only store opening angle (2D texture suffices!)





# Links

- Overview

[gbitscience.blogspot.de/2013/07/raymarching-distance-fields\\_14.html](http://gbitscience.blogspot.de/2013/07/raymarching-distance-fields_14.html)

- Distance functions

[www.iquilezles.org/www/articles/distfunctions/distfunctions.htm](http://www.iquilezles.org/www/articles/distfunctions/distfunctions.htm)