

# Synthèse d'Images

de l'immersif au photo-realiste

Loic Simon

[www.ensicaen.fr](http://www.ensicaen.fr)

GREYC  
image

# Synthèse d'Images

A quoi cela sert-il ?

# Jeux vidéos / cinéma



# Jeux videos / cinéma



more awesome pictures at [THEMETAPICTURE.COM](http://THEMETAPICTURE.COM)

# Jeux vidéos / cinéma



# Jeux vidéos / cinéma



# Jeux vidéos / cinéma



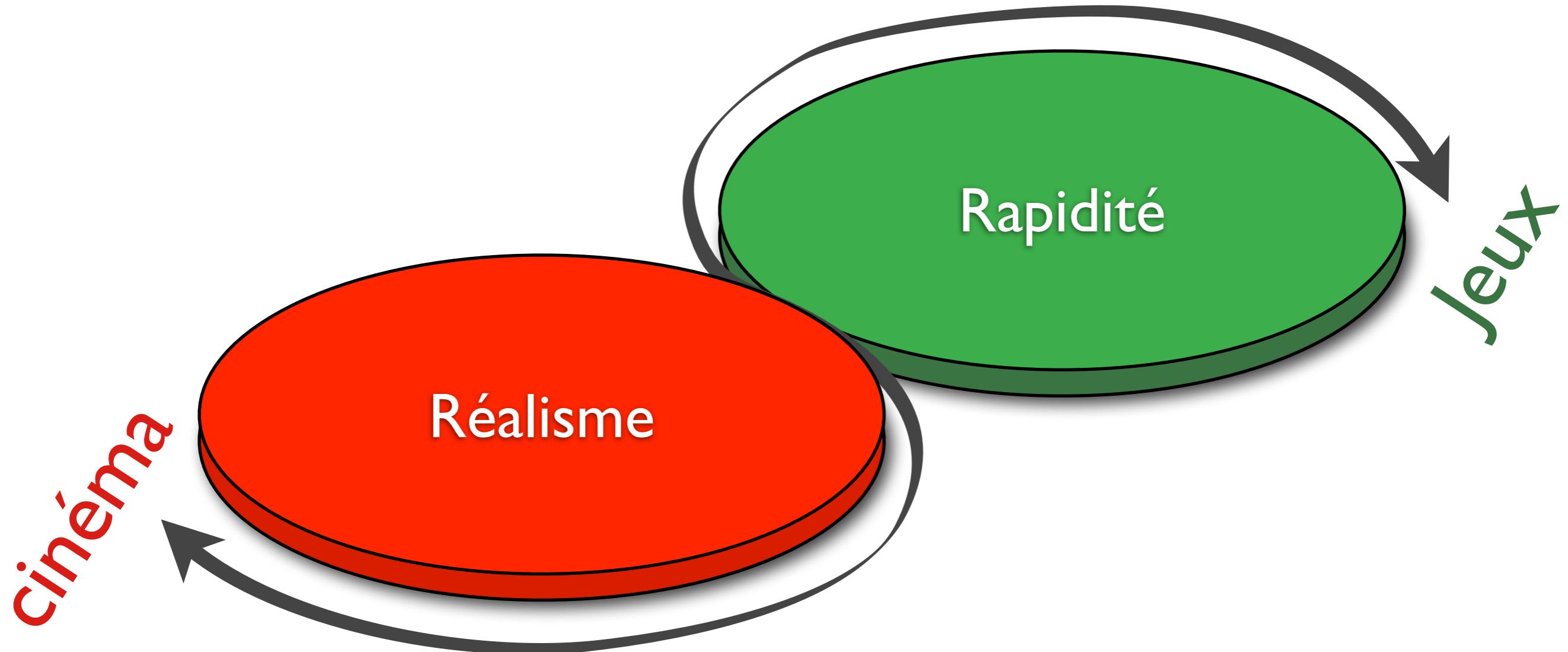
# Jeux vidéos / cinéma



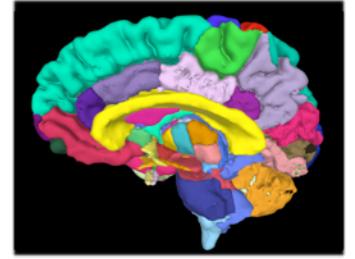
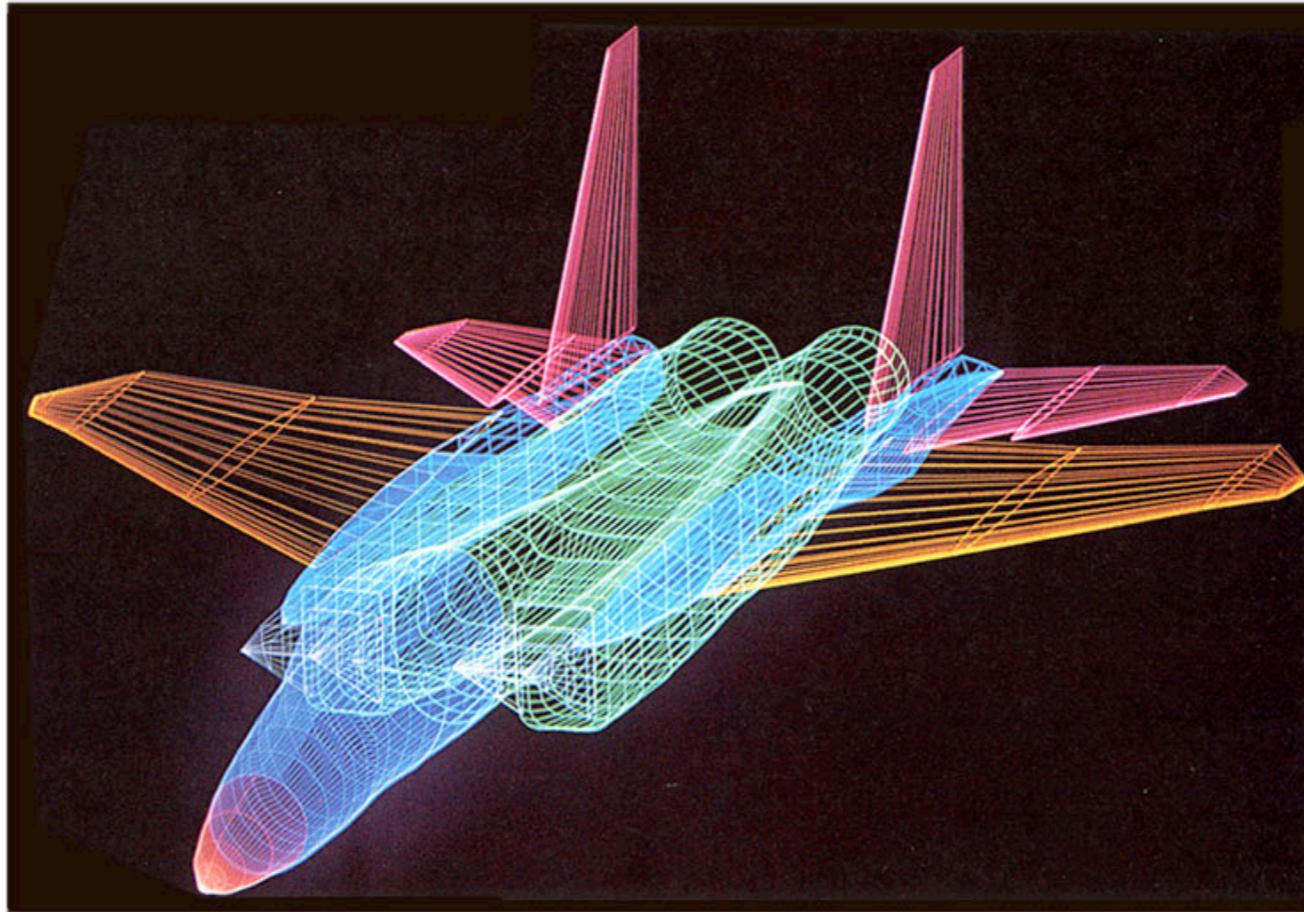
# Jeux vidéos / cinéma



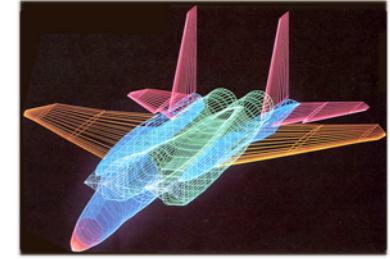
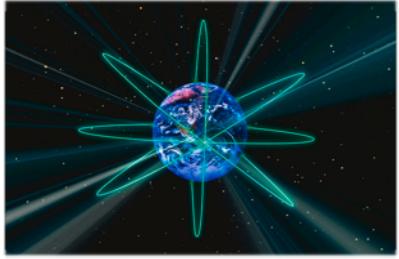
# Enjeux



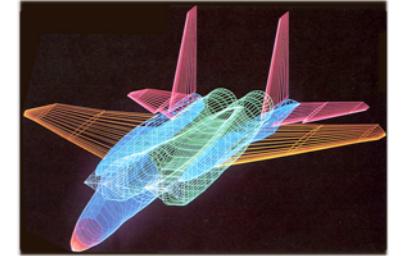
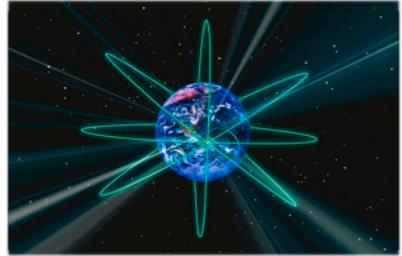
# Scientific visualization



# Scientific visualization



# Scientific visualization



“Mountains are not cones, clouds are not spheres, trees are not cylinders, ...”

B. Mandelbrot

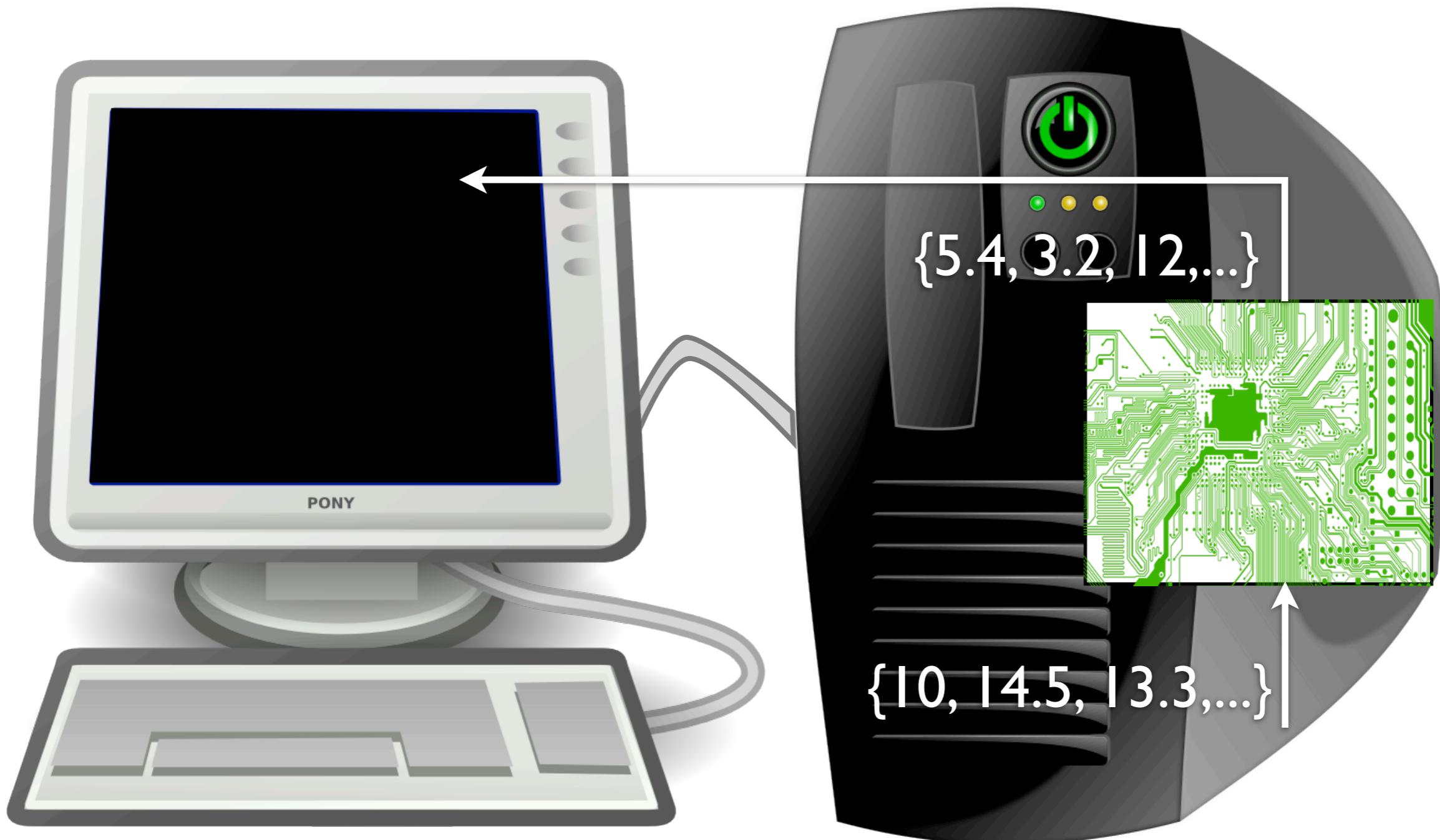
# Synthèse d'Images

## Comment ça marche ?

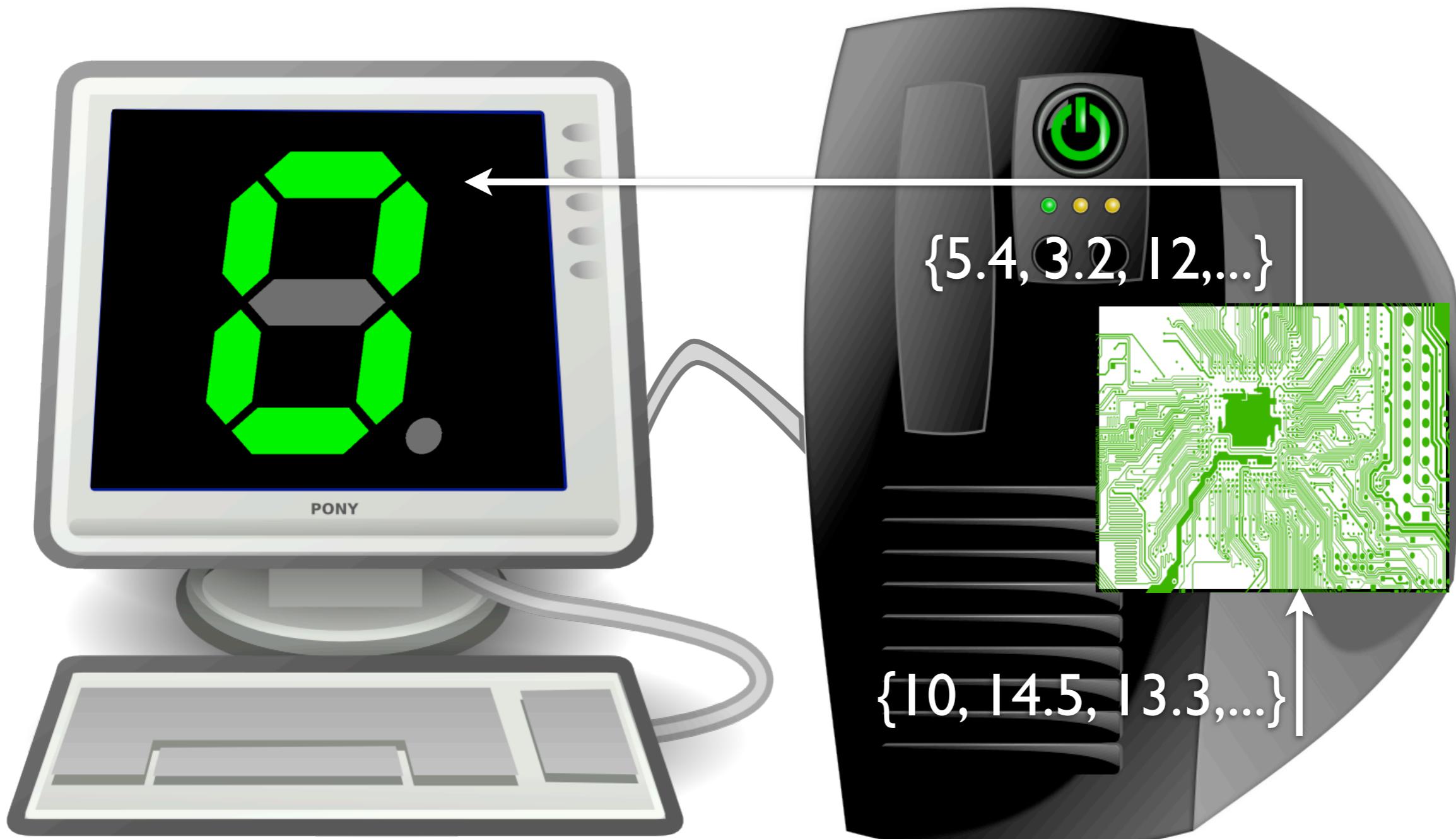
# Aspects numériques



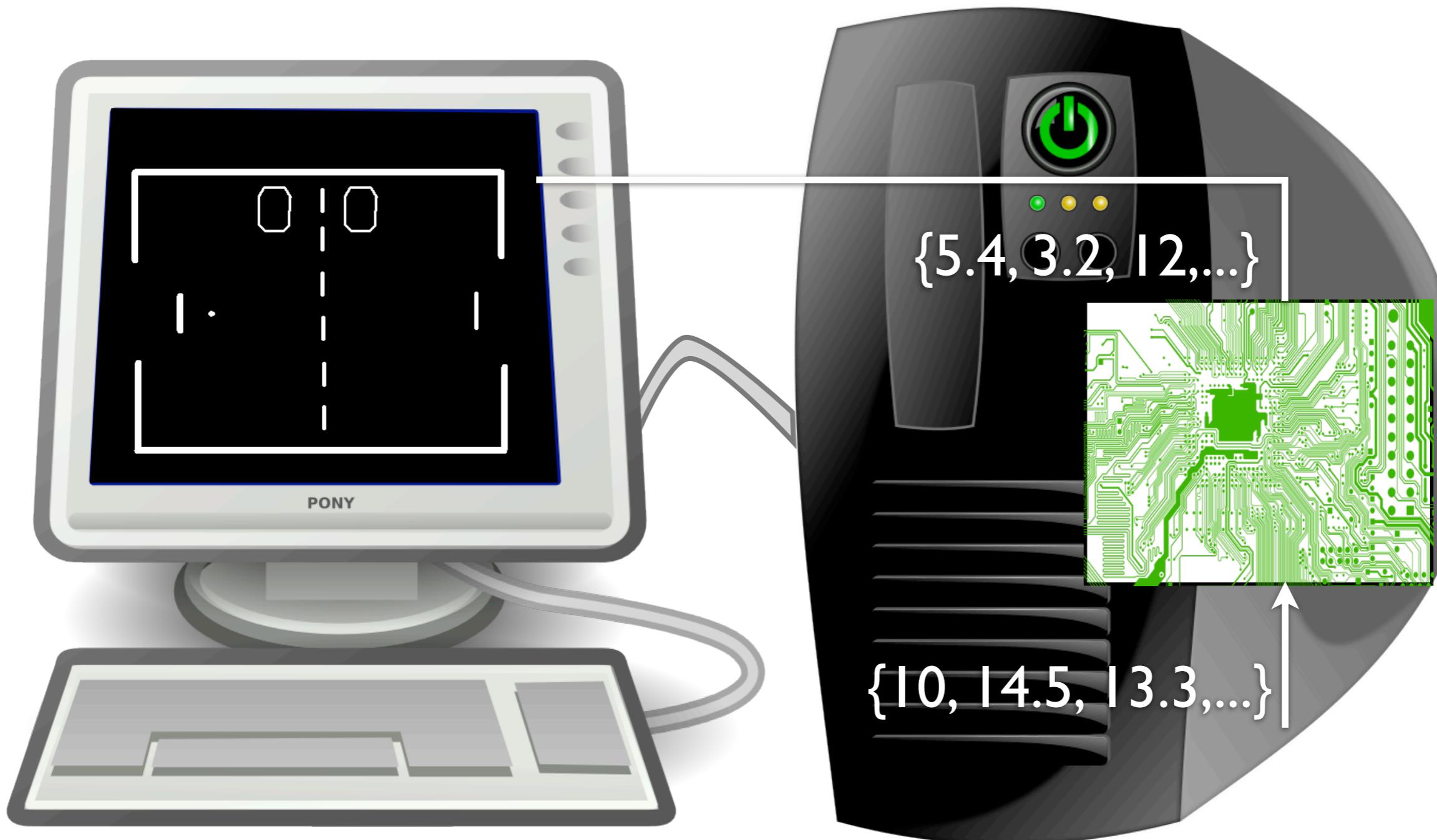
# Aspects numériques



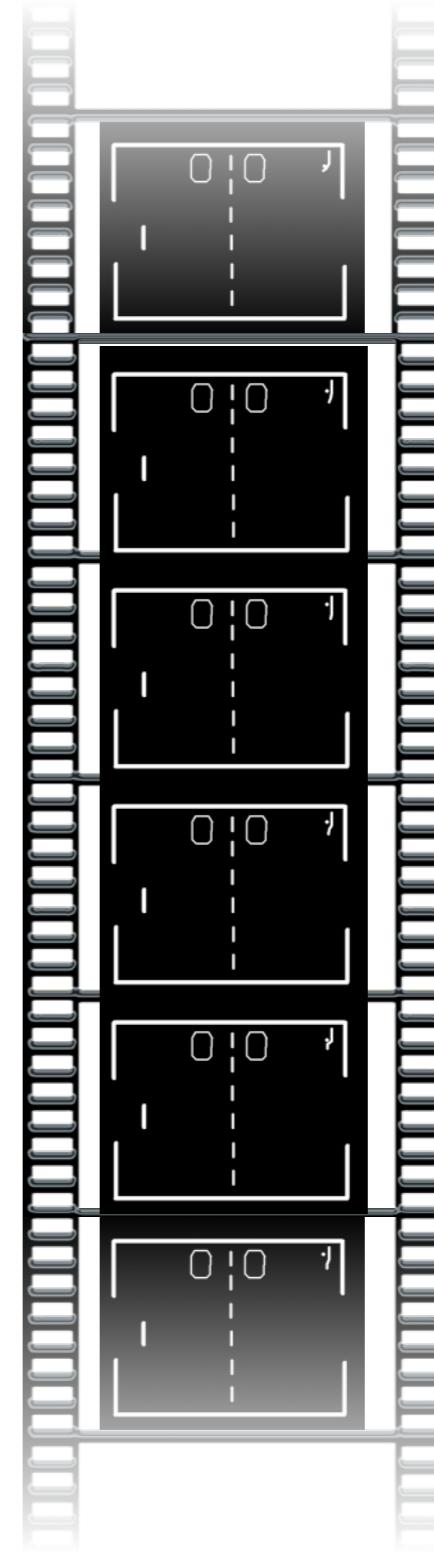
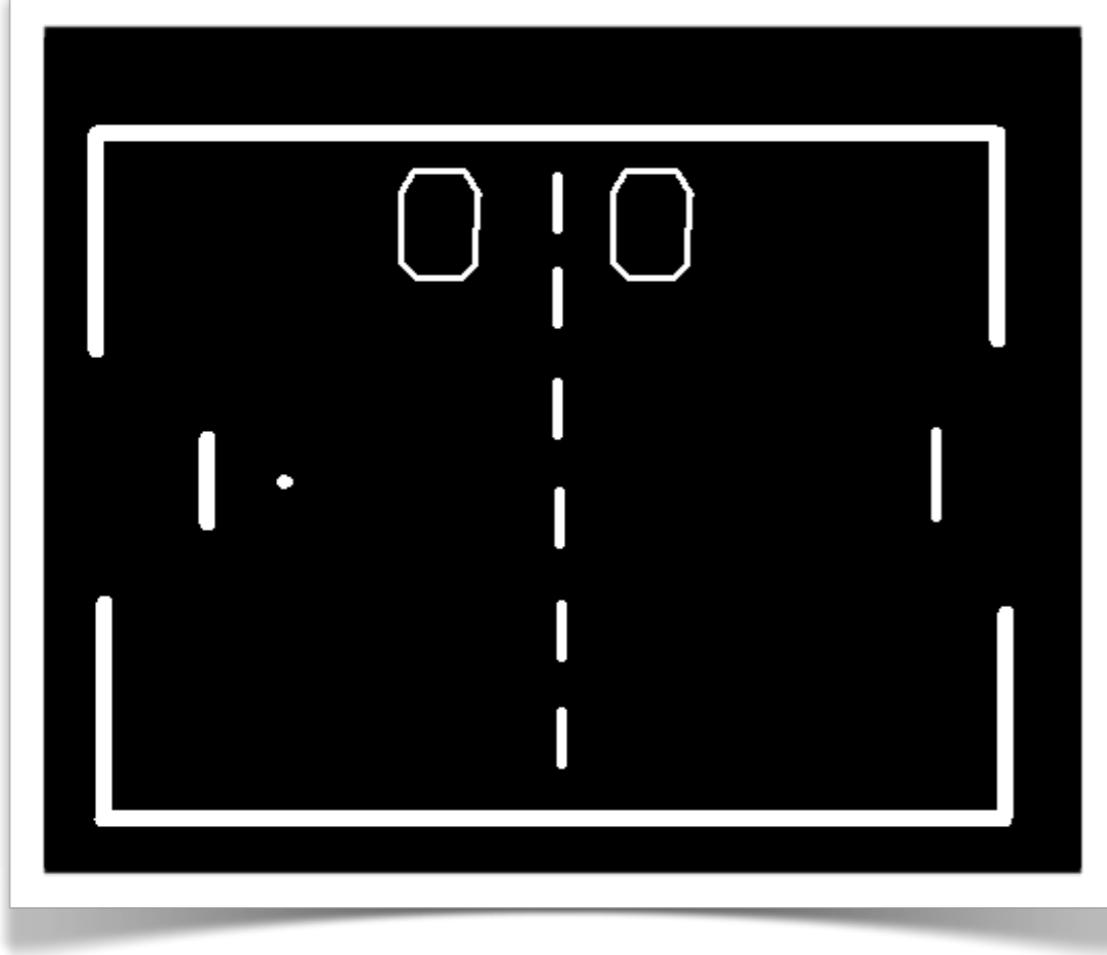
# Aspects numériques



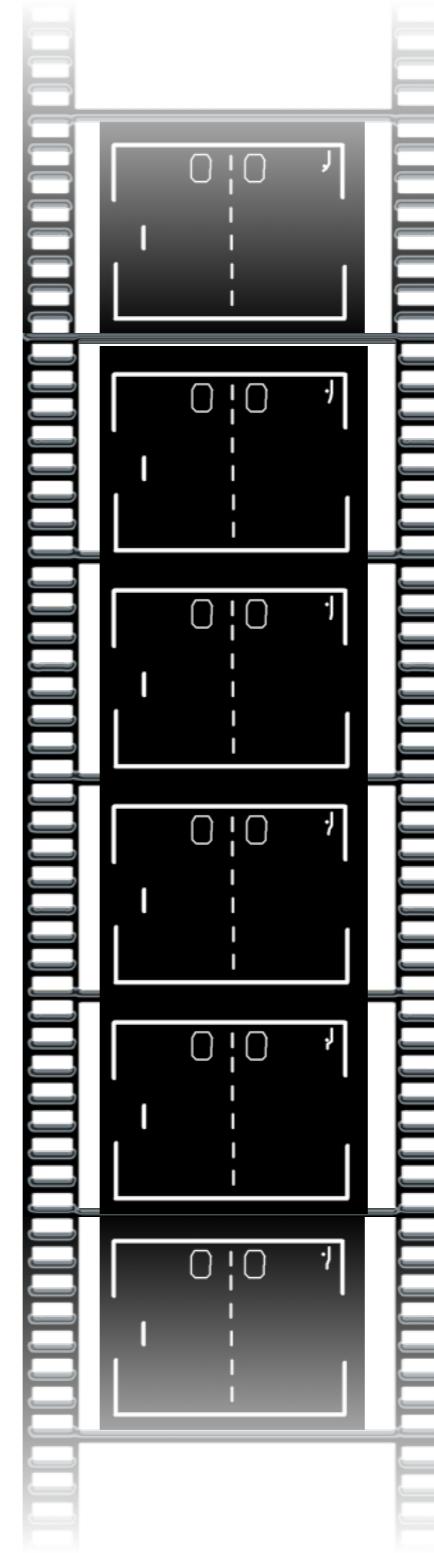
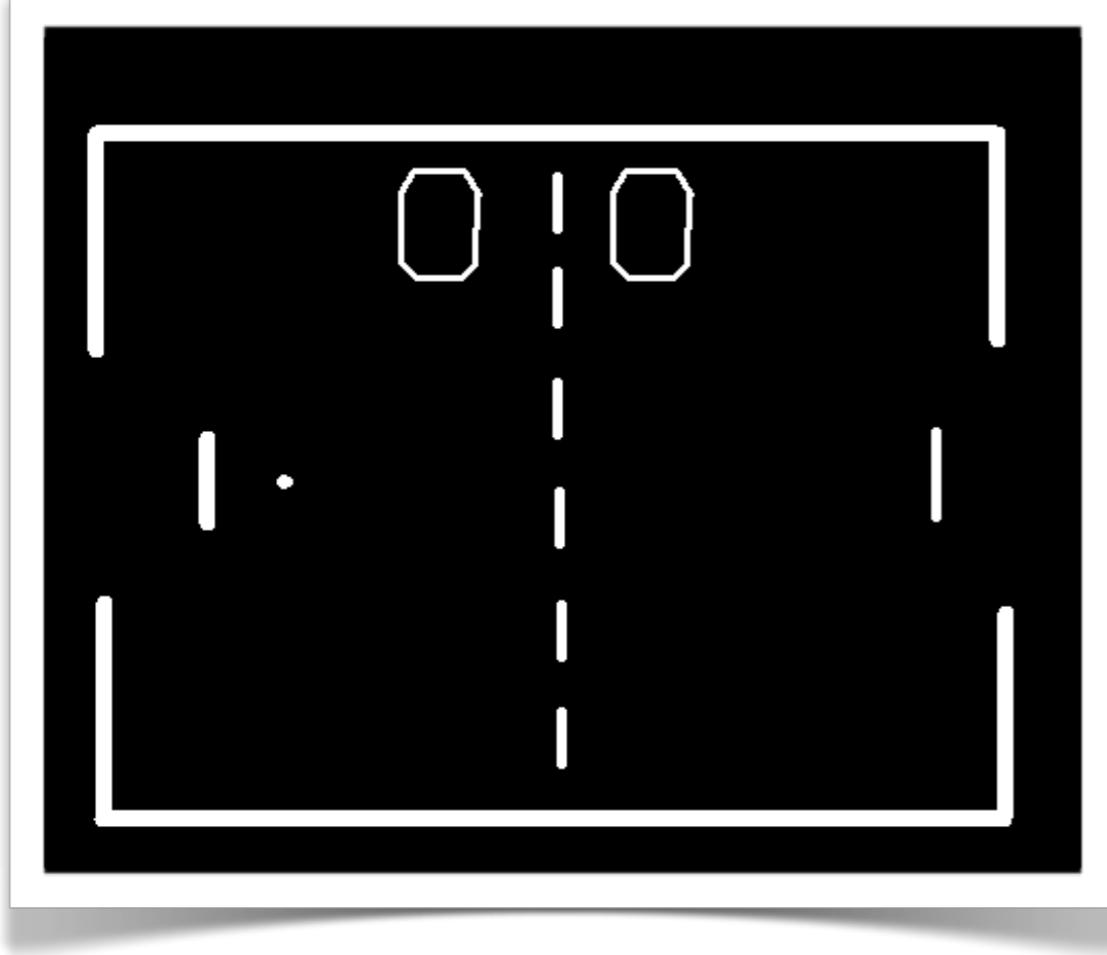
# Aspects numériques



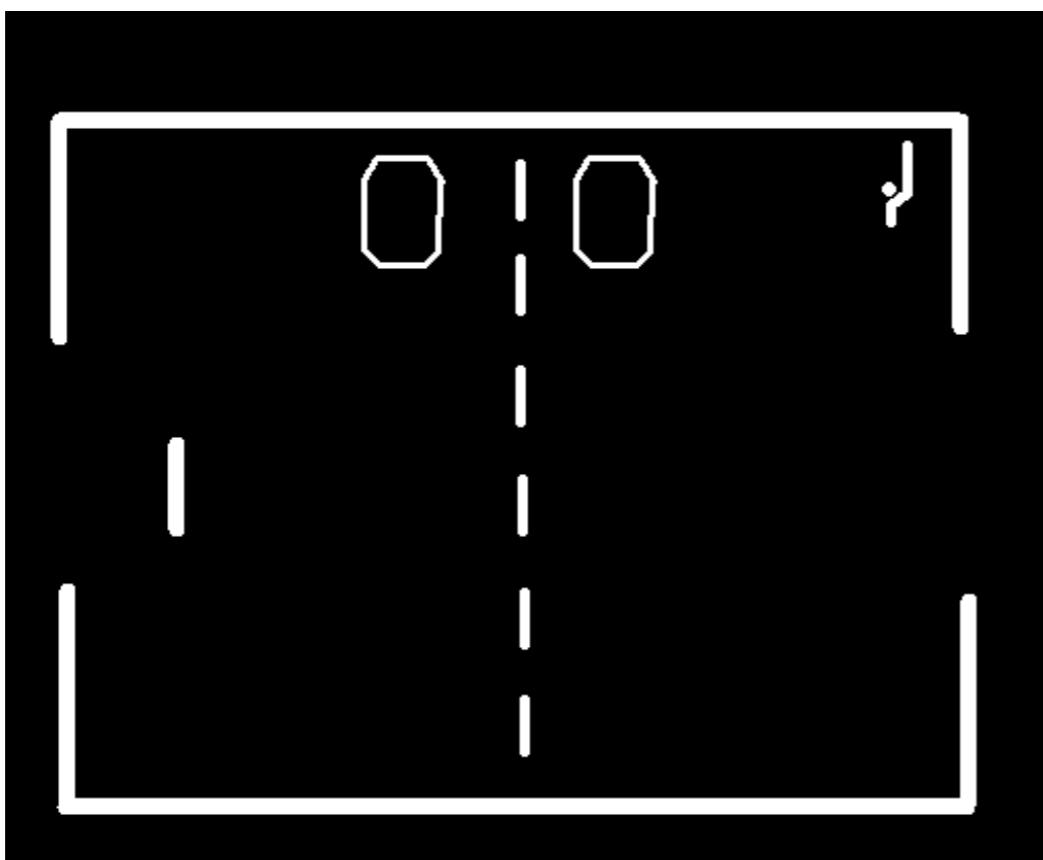
# Animation et frames



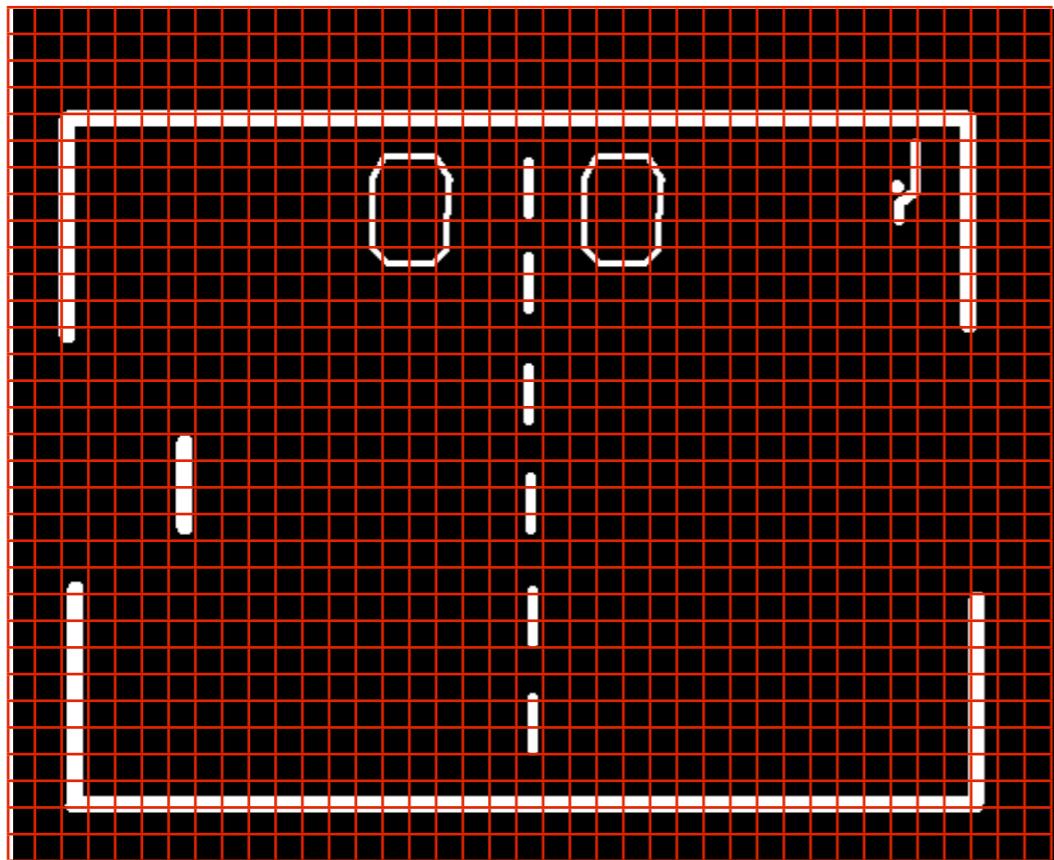
# Animation et frames



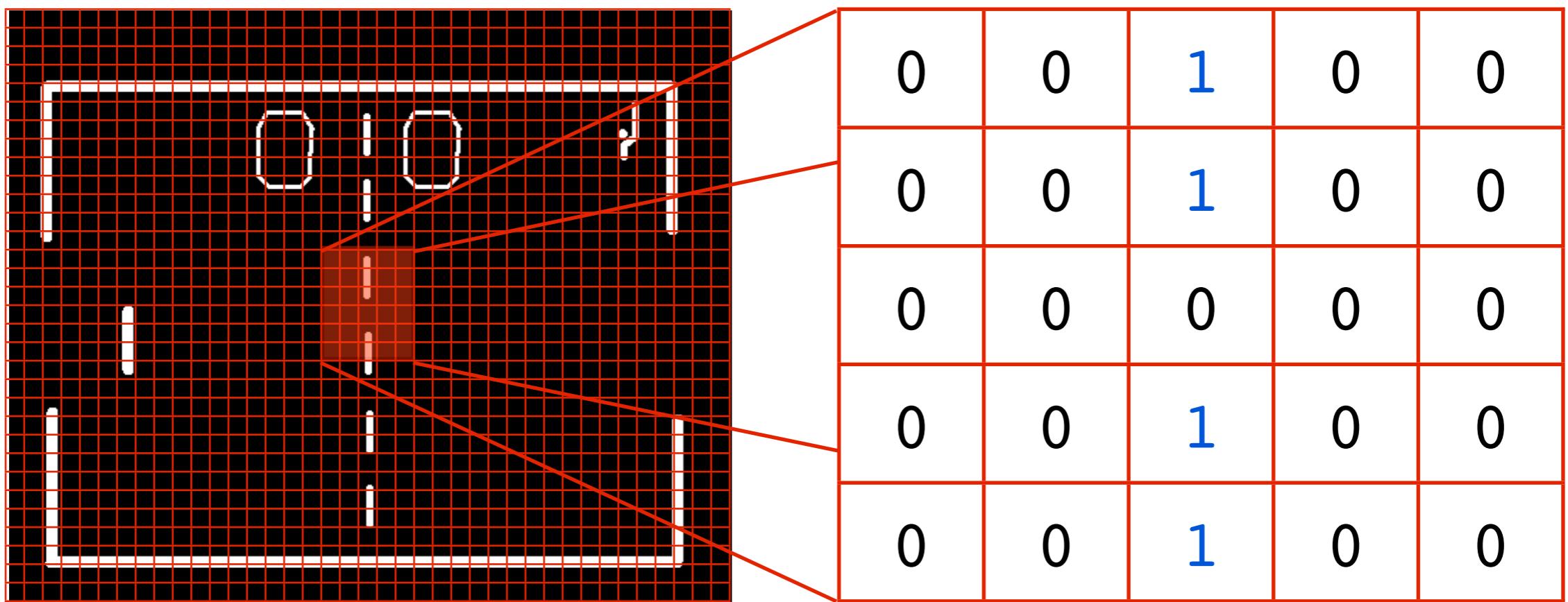
# Frame et pixels



# Frame et pixels



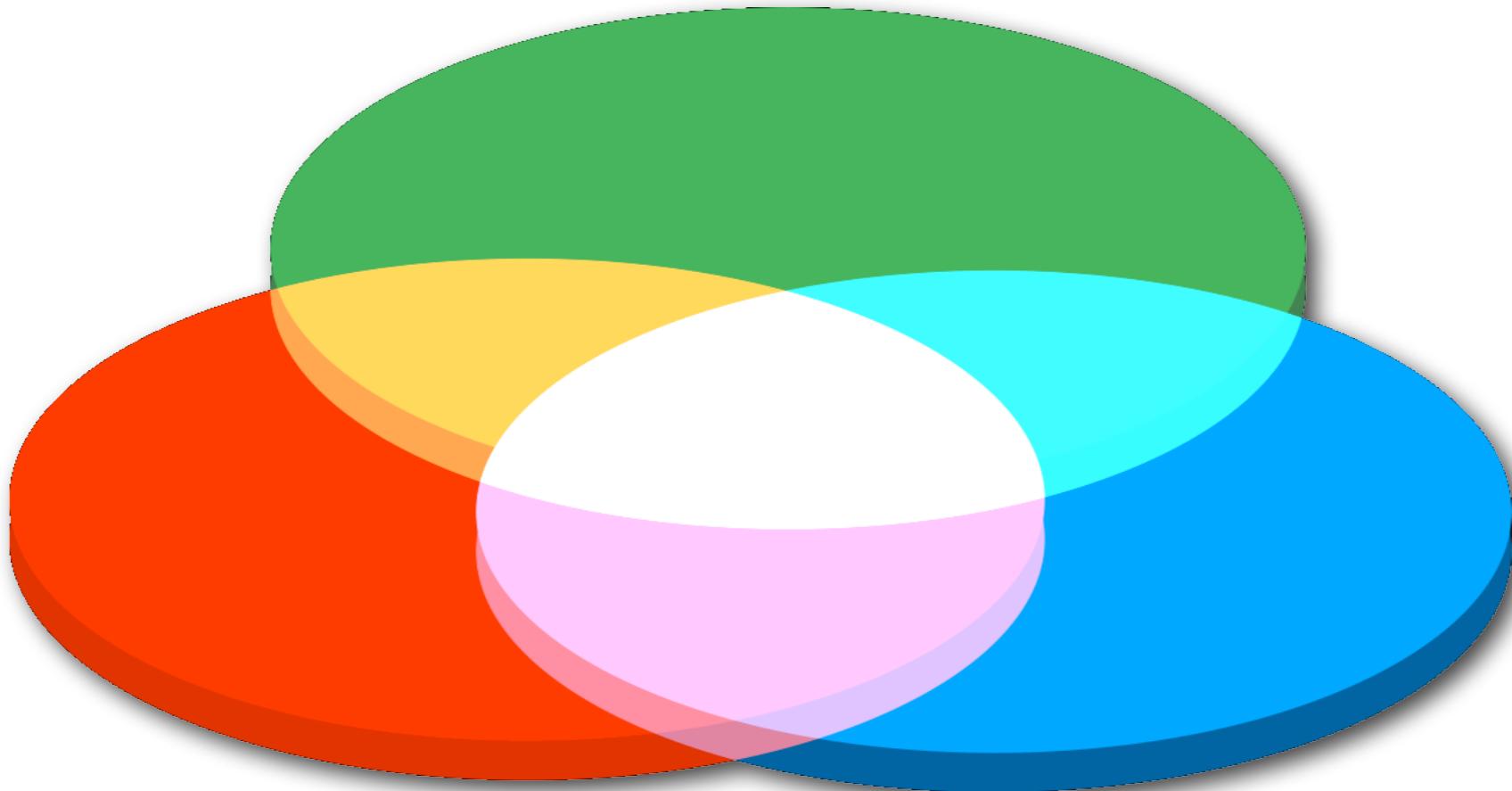
# Frame et pixels



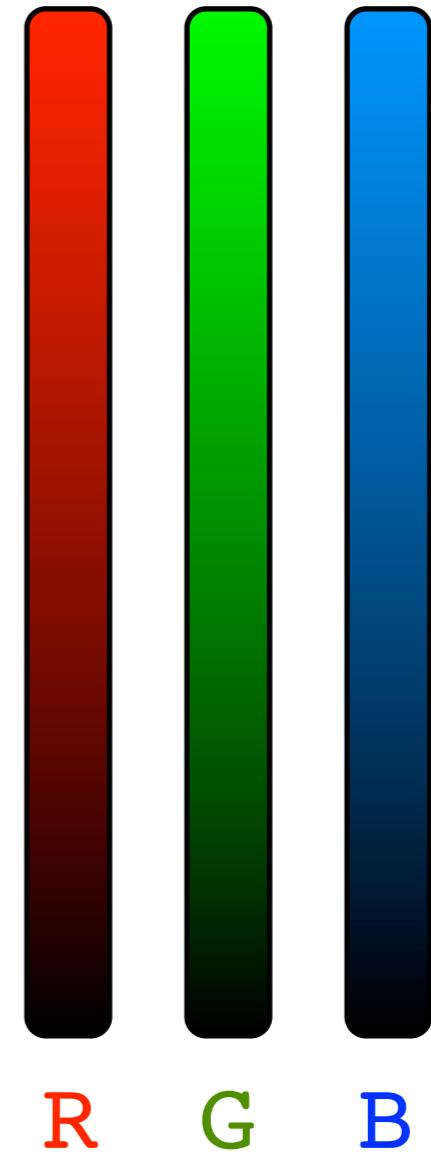
# Encodage des pixels



# Synthèse additive



# Encodage des pixels



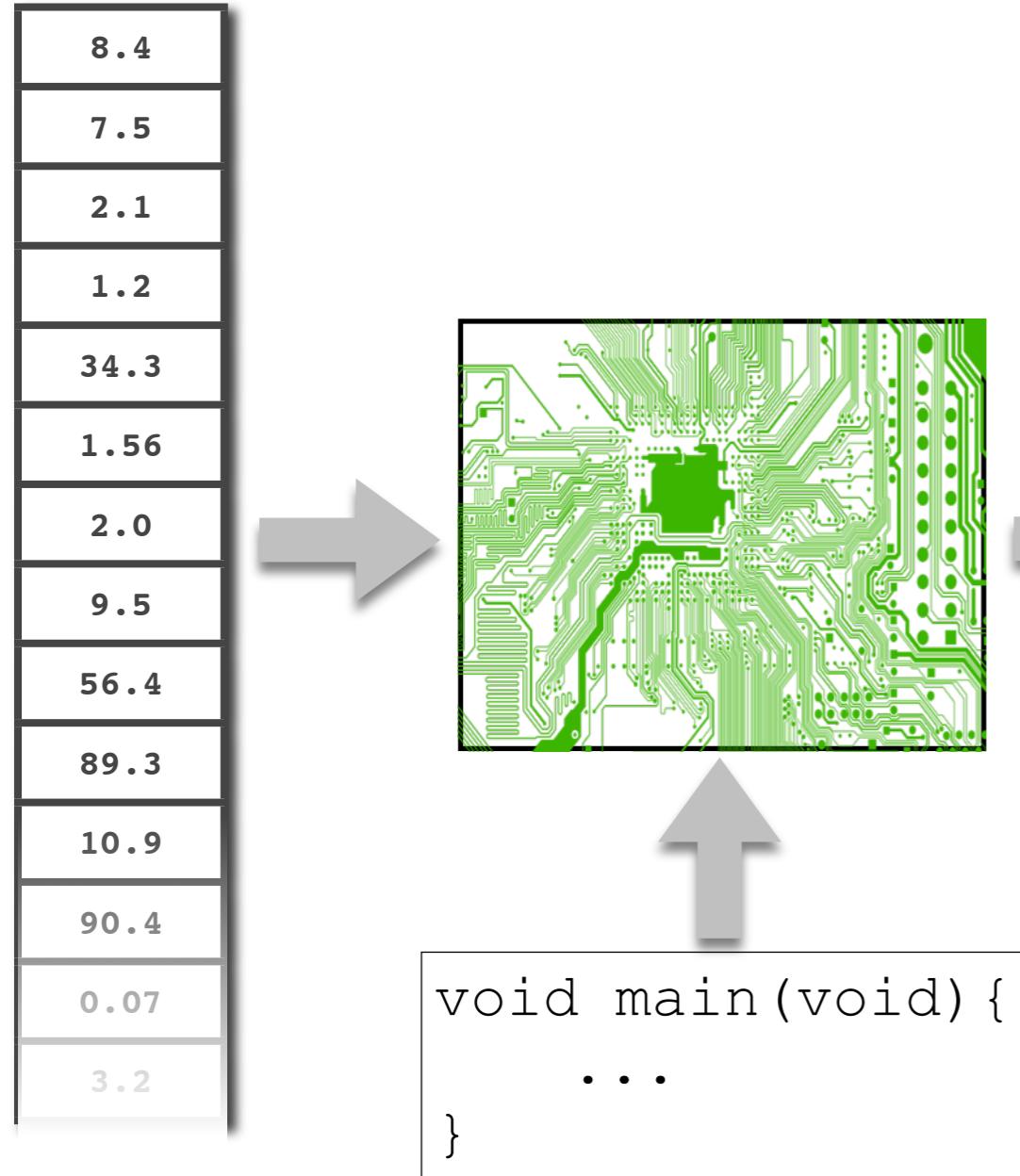
# Encodage d'une frame



2

97	8	1	8	1
43				
78				
3	1			
47	3			
65				
54	2			
64	34	1	98	2
45	1	95	19	21
6	3	35	7	9
90	0	8	4	8
7	12	1	43	56
32	1	32	89	

# CGI pipeline



Représentation

Algorithme

Frame(s)

# CGI Objective



👉 Simuler la photographie d'un objet 3D

# CGI Objective



☞ Simuler la photographie d'un objet 3D

# CGI Objective

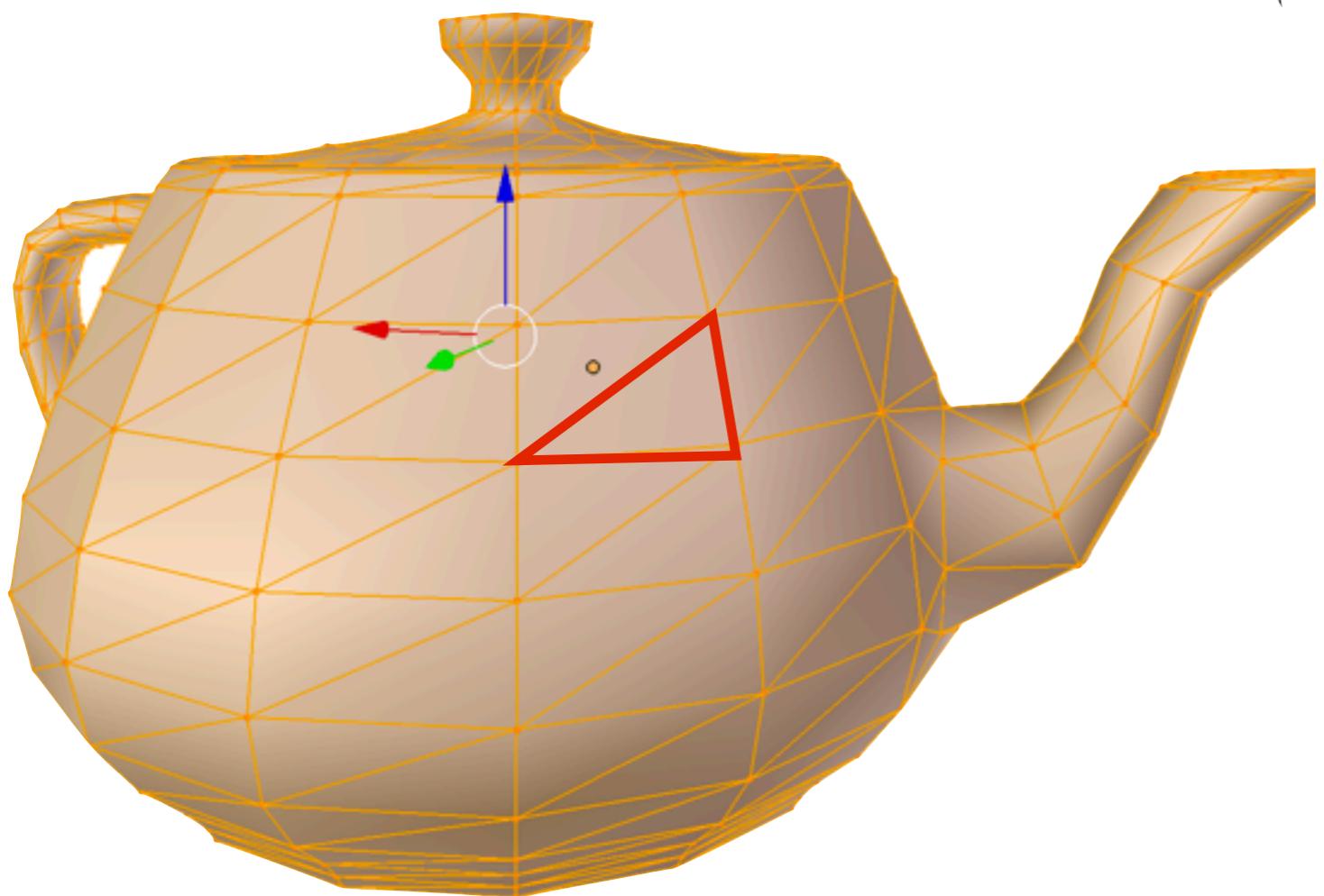


☞ Simuler la photographie d'un **objet 3D**

# Représentation 3D

Maillages, textures ...

# Maillages



Triangle 0

Triangle 1

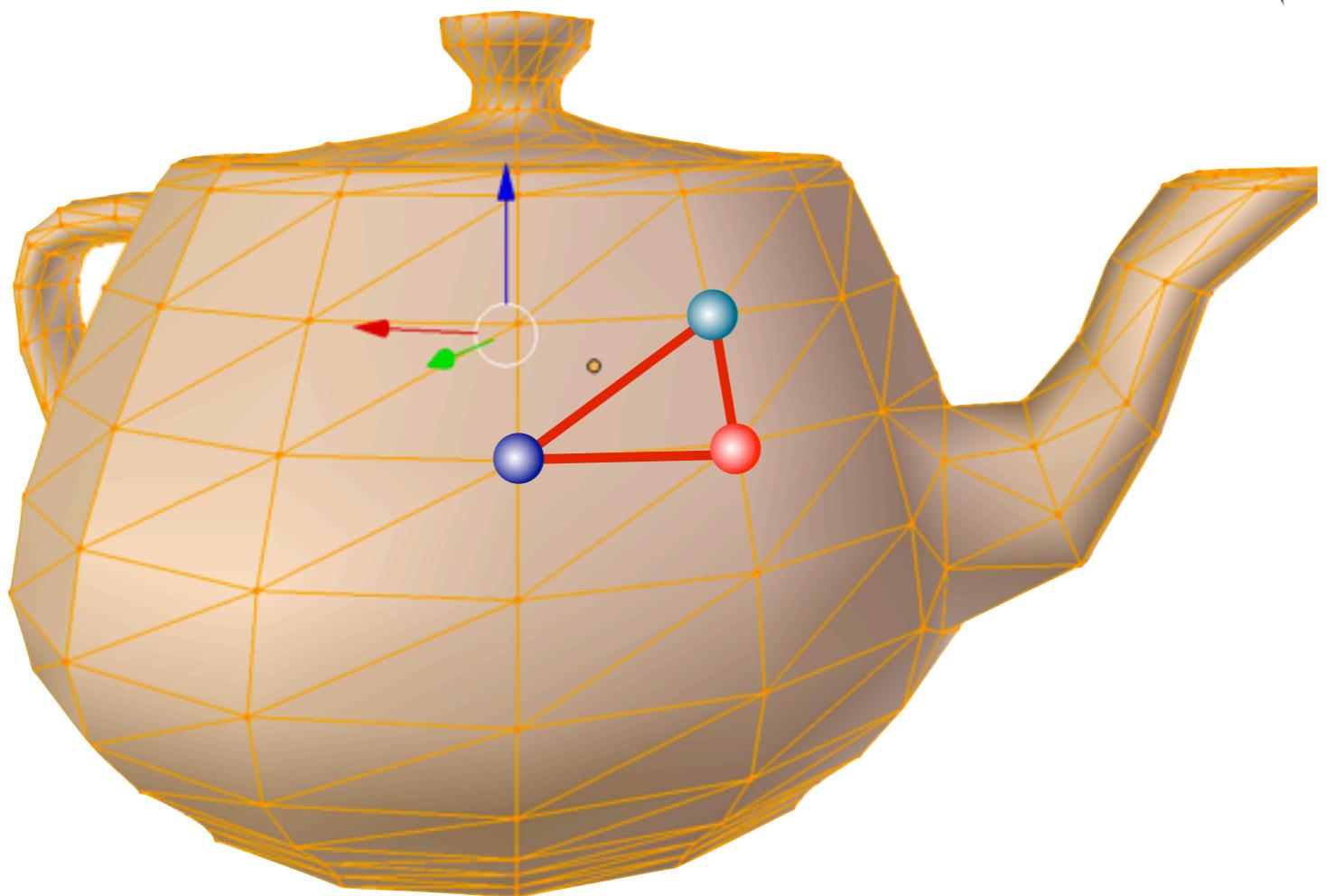
Triangle 2

Triangle 3

Triangle 4

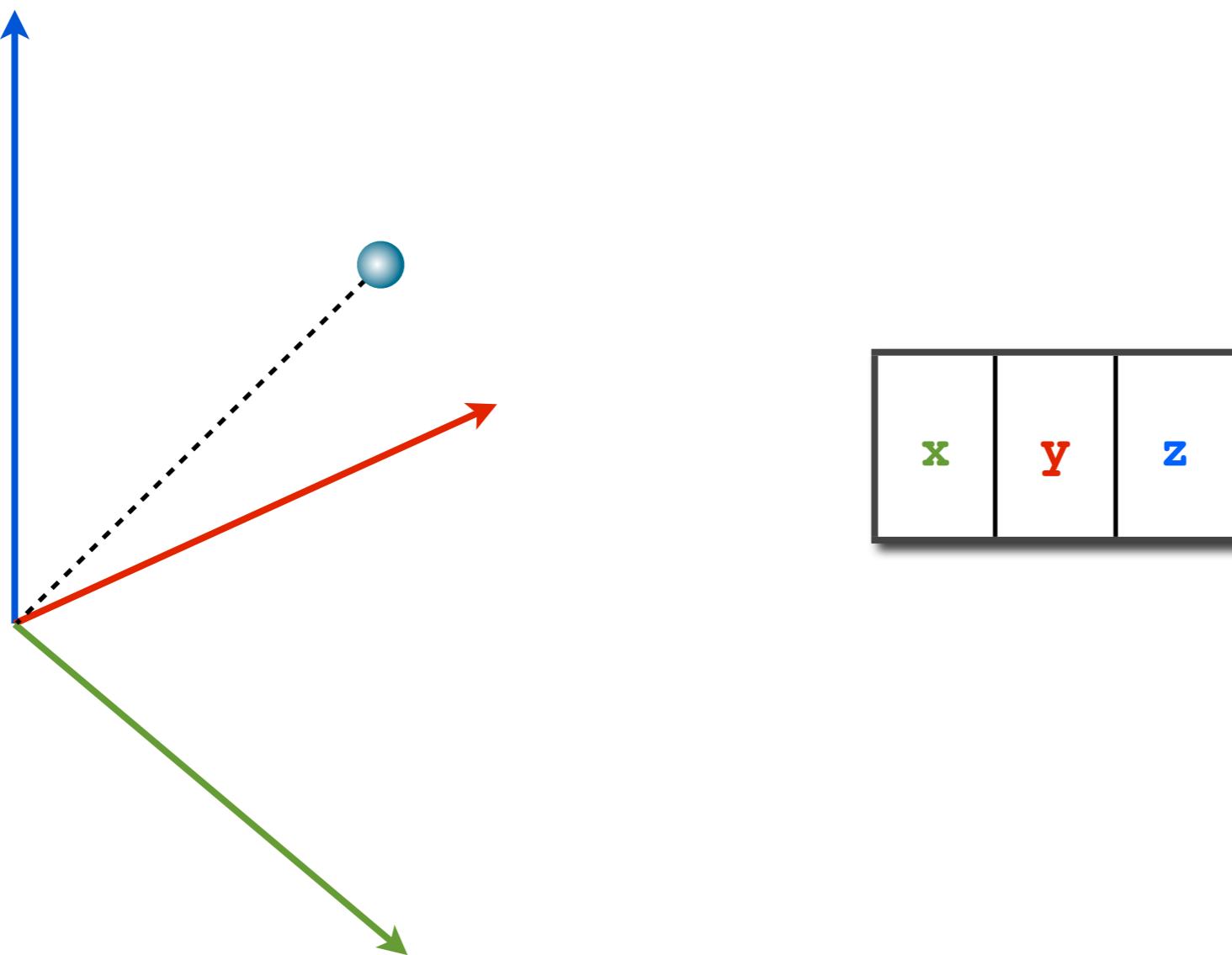
Triangle 5

# Maillages

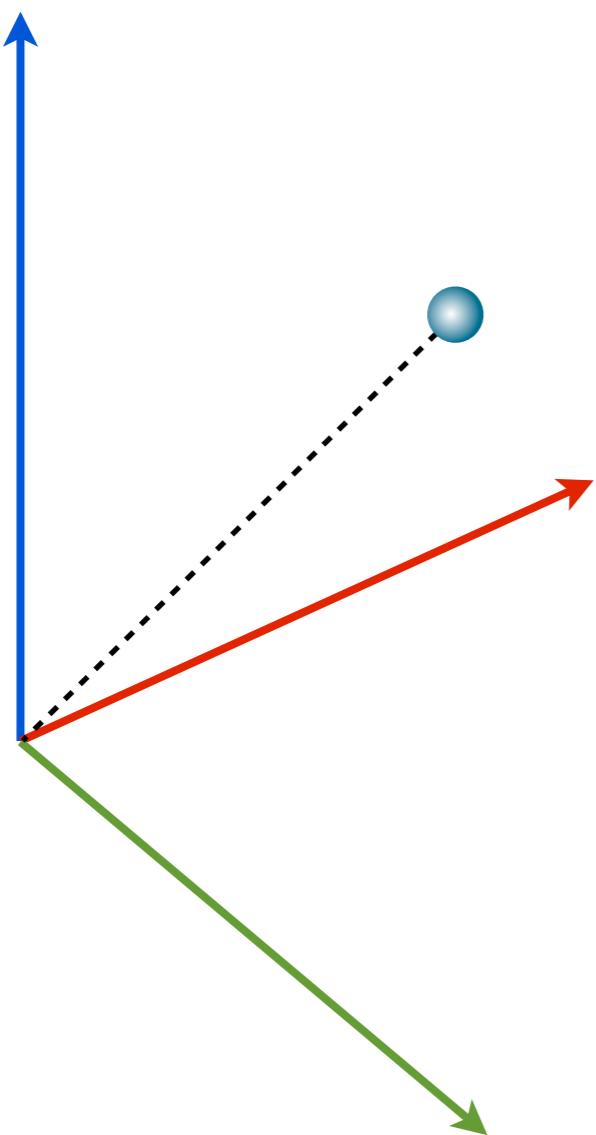


v0	v1	v2
v3	v4	v5
v6	v2	v1
v4	v3	v7
v8	v9	v5
v9	v5	v6

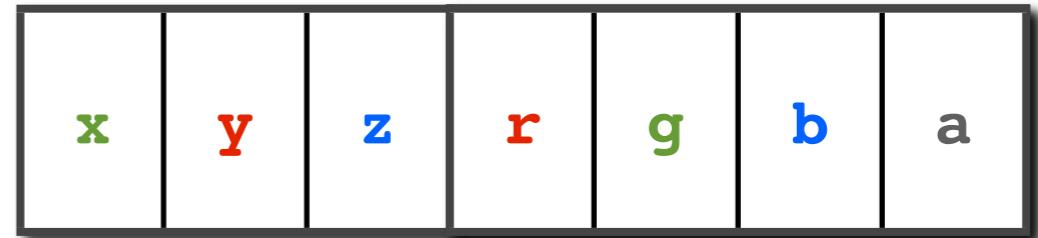
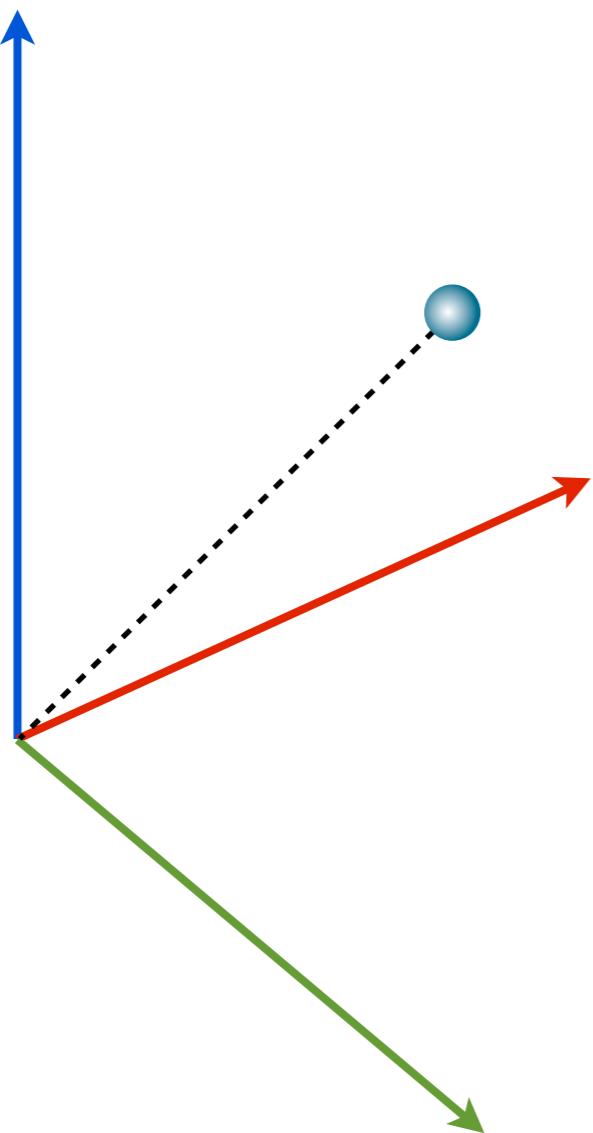
# Vertex



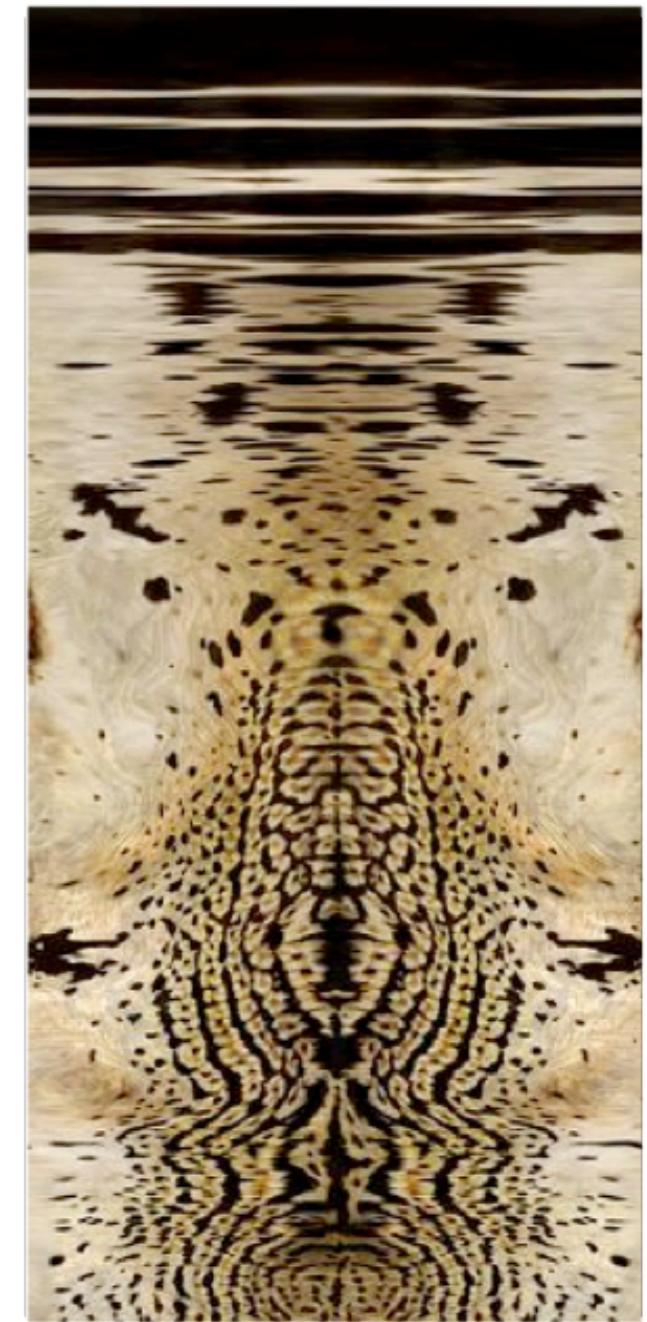
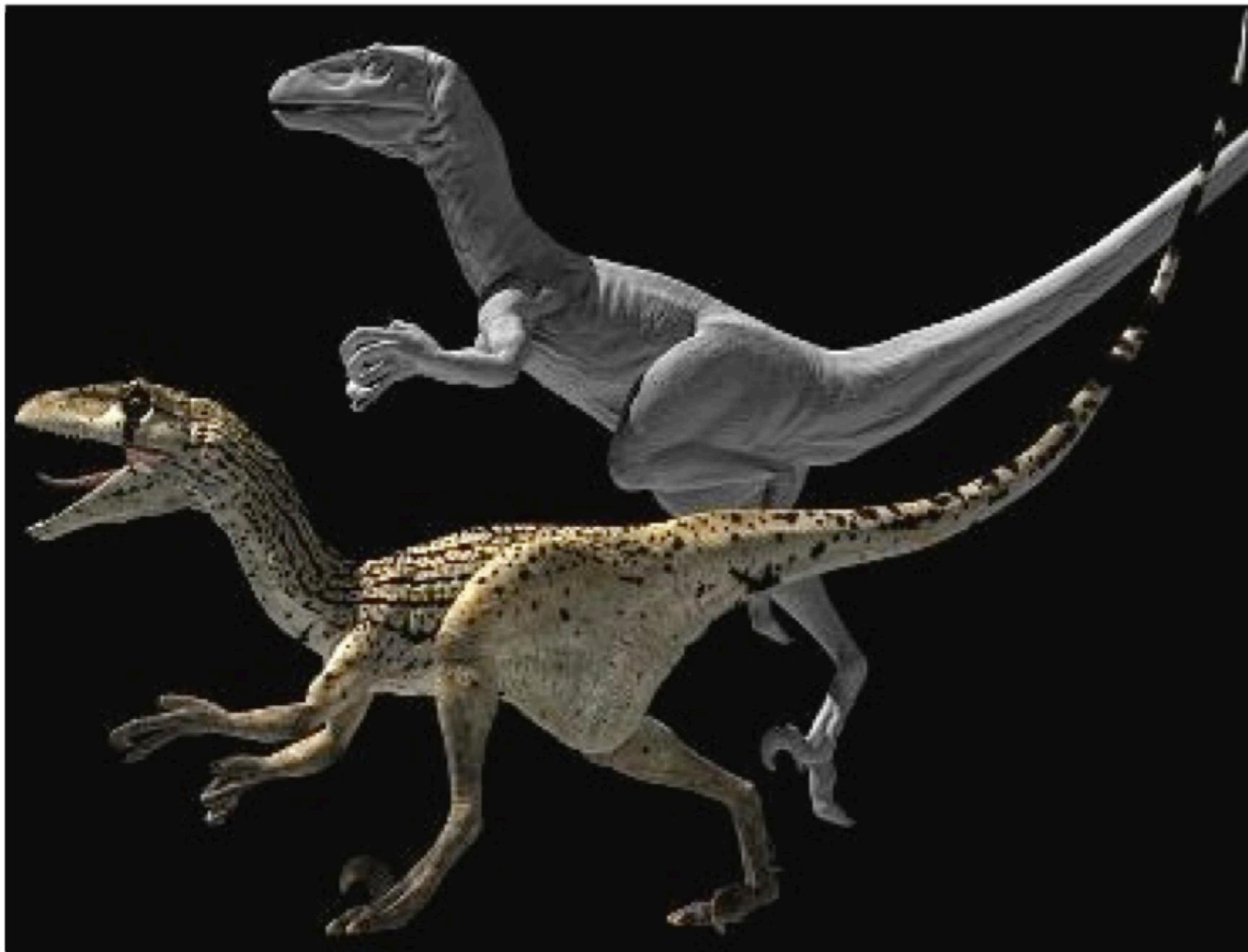
# Vertex



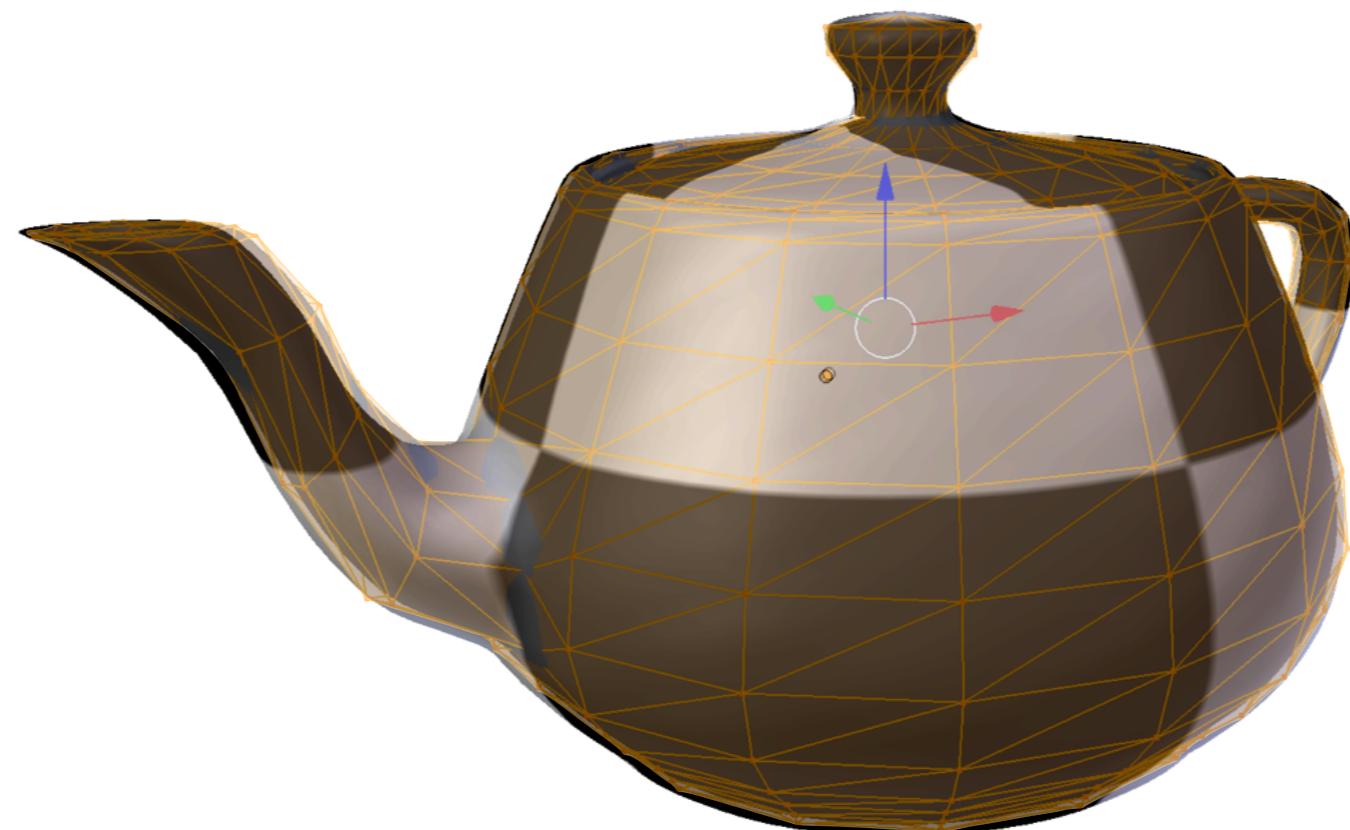
# Vertex



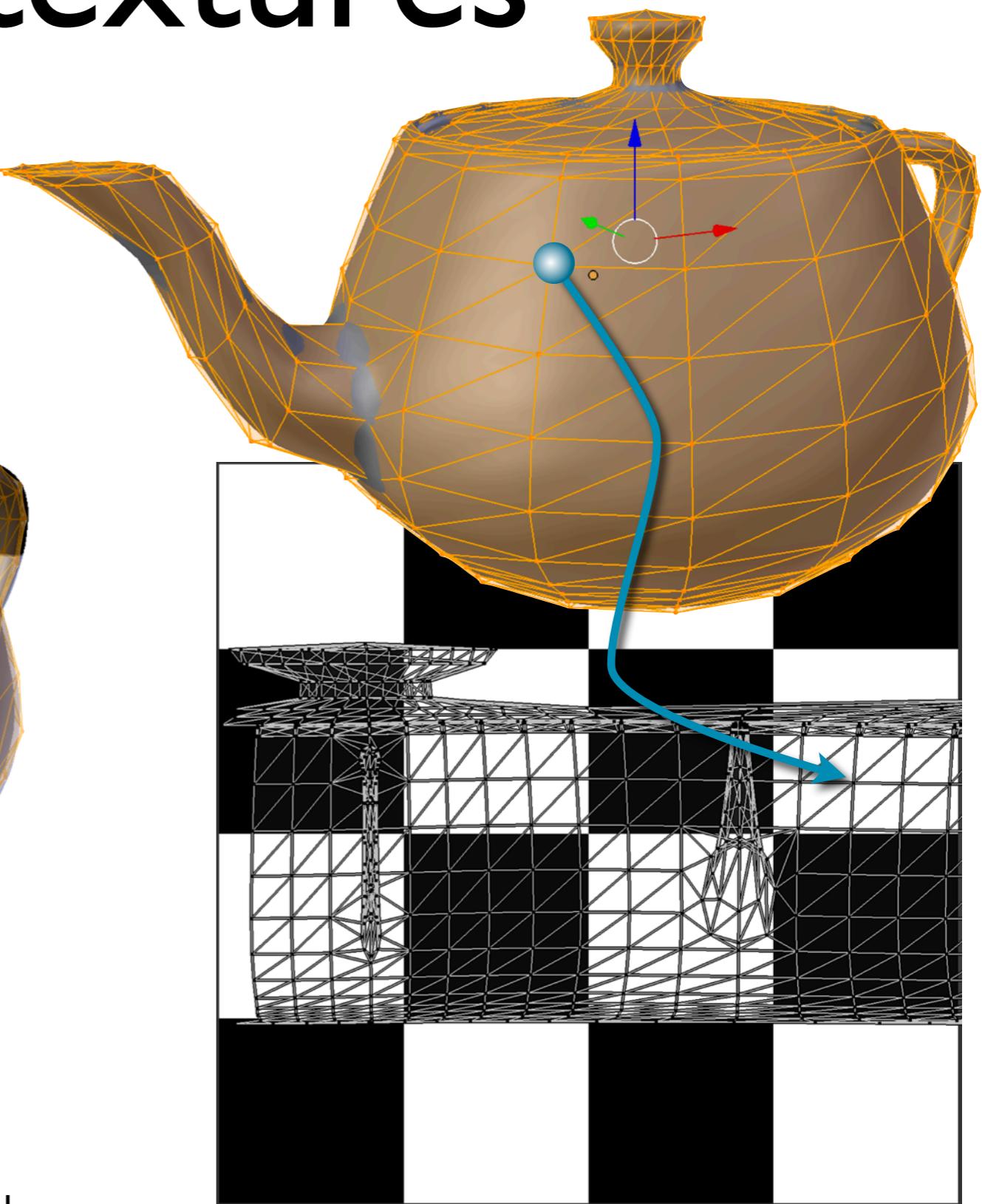
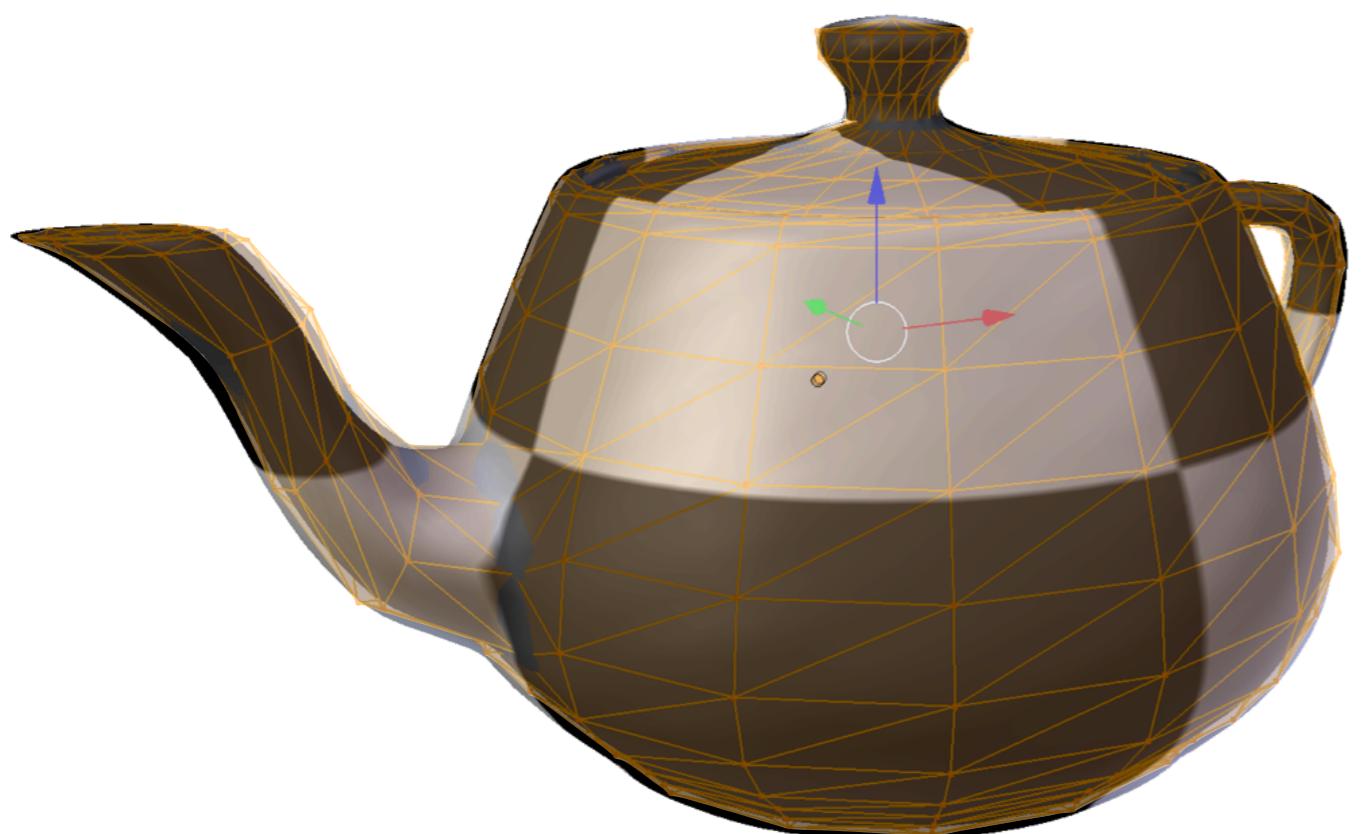
# Objets texturés



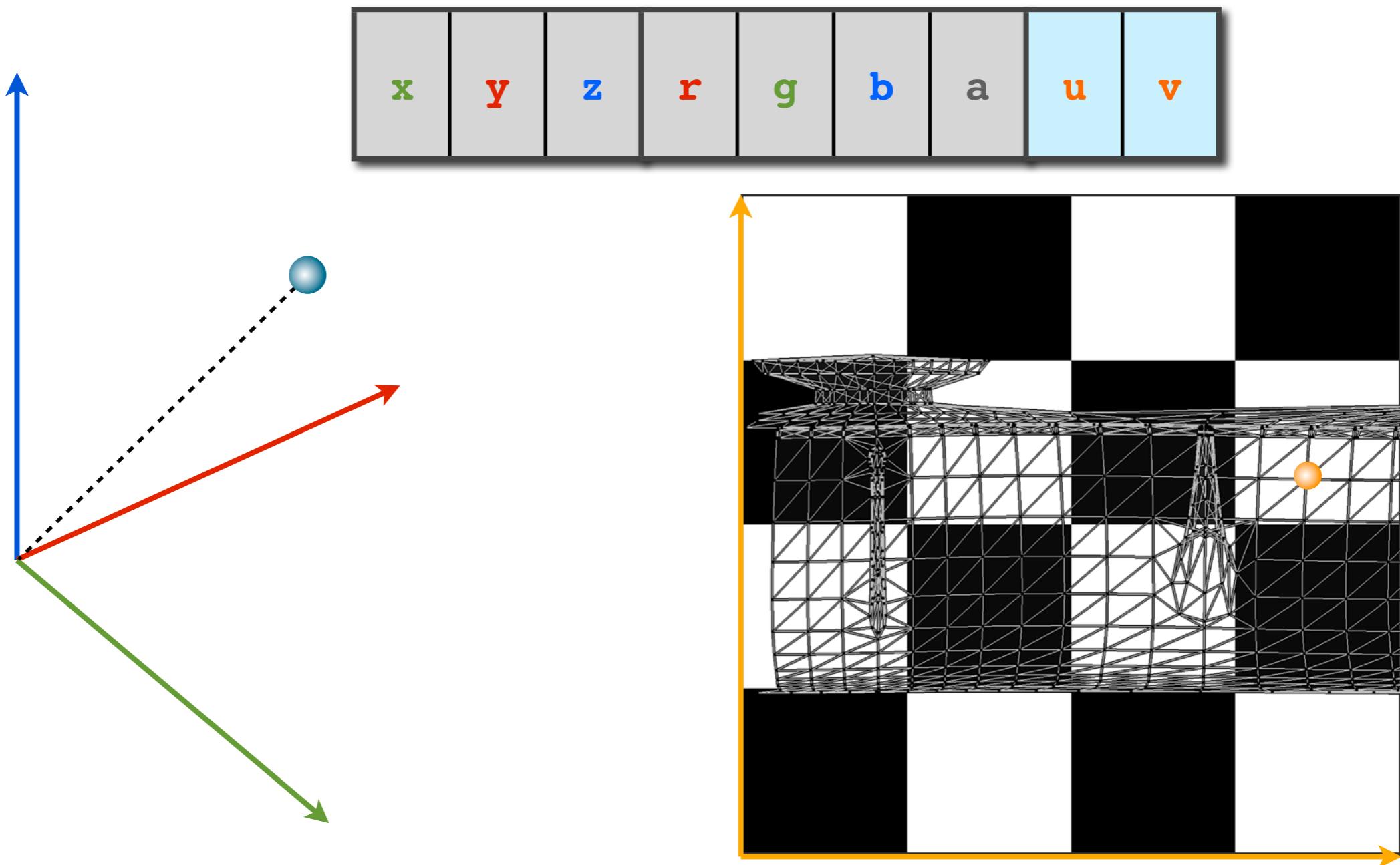
# Objets texturés



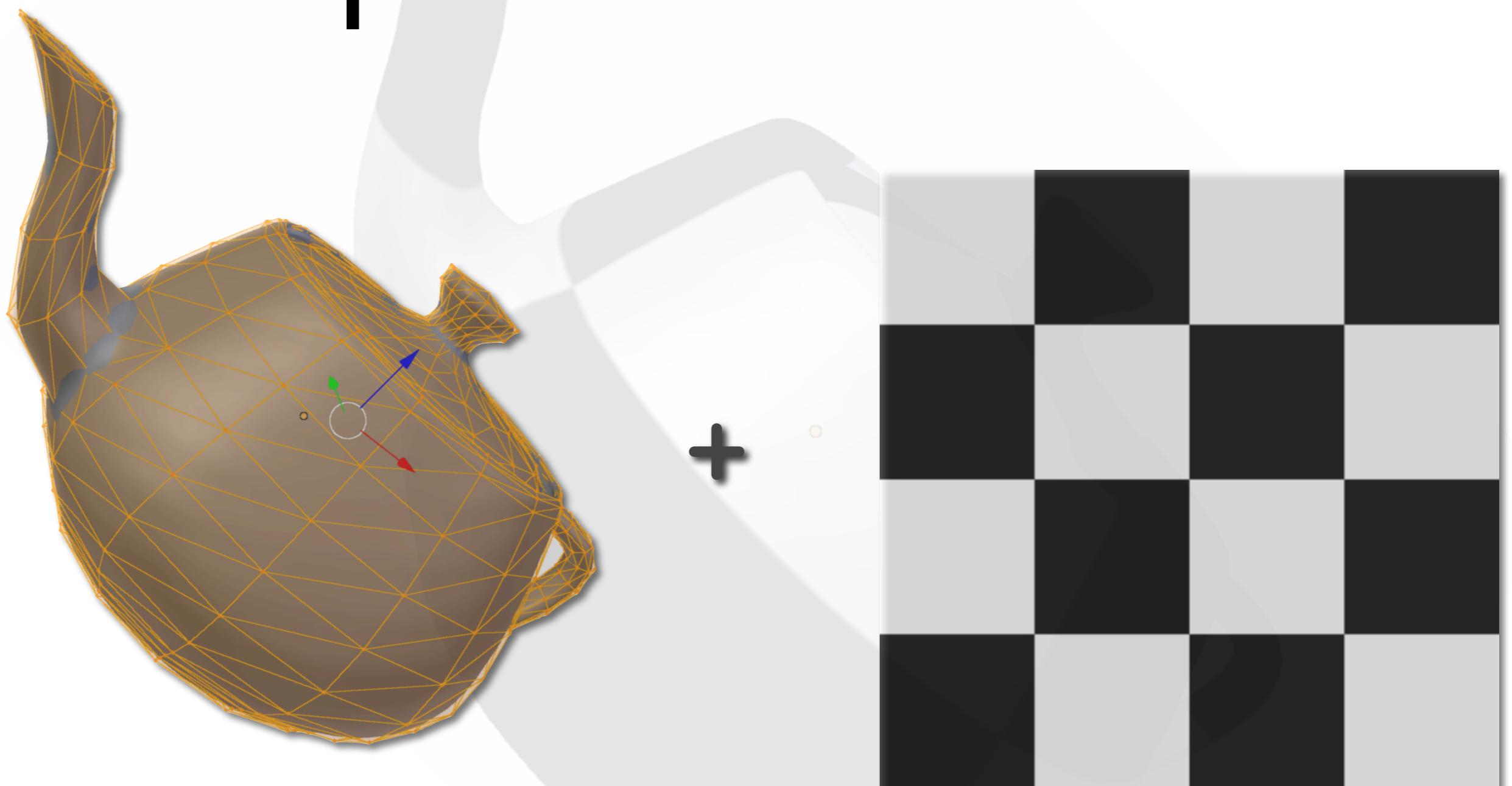
# Objets texturés



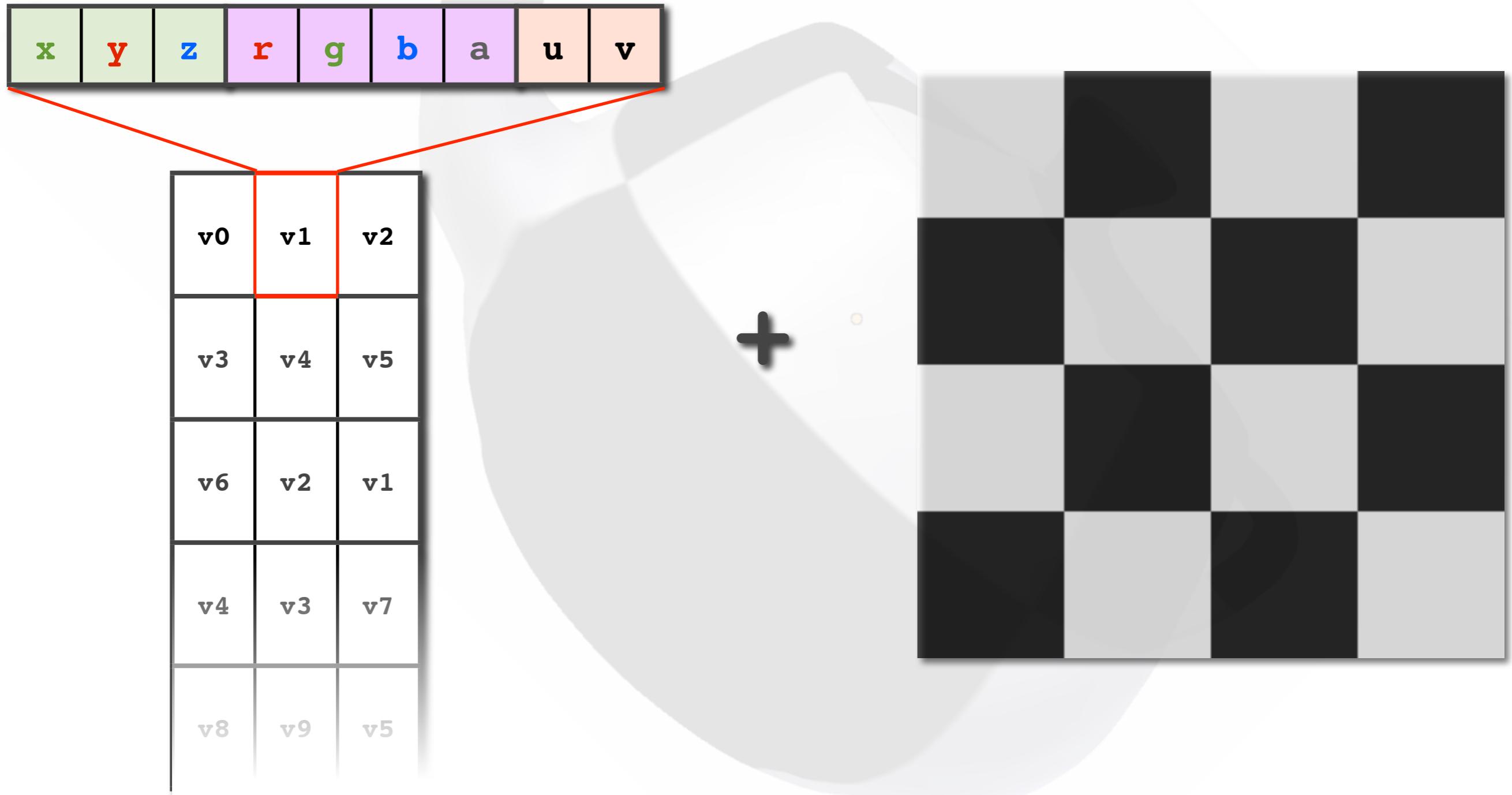
# UV mapping



# Représentation 3D



# Représentation 3D



# Représentation 3D

x	y	z	r	g	b	a	u	v
---	---	---	---	---	---	---	---	---



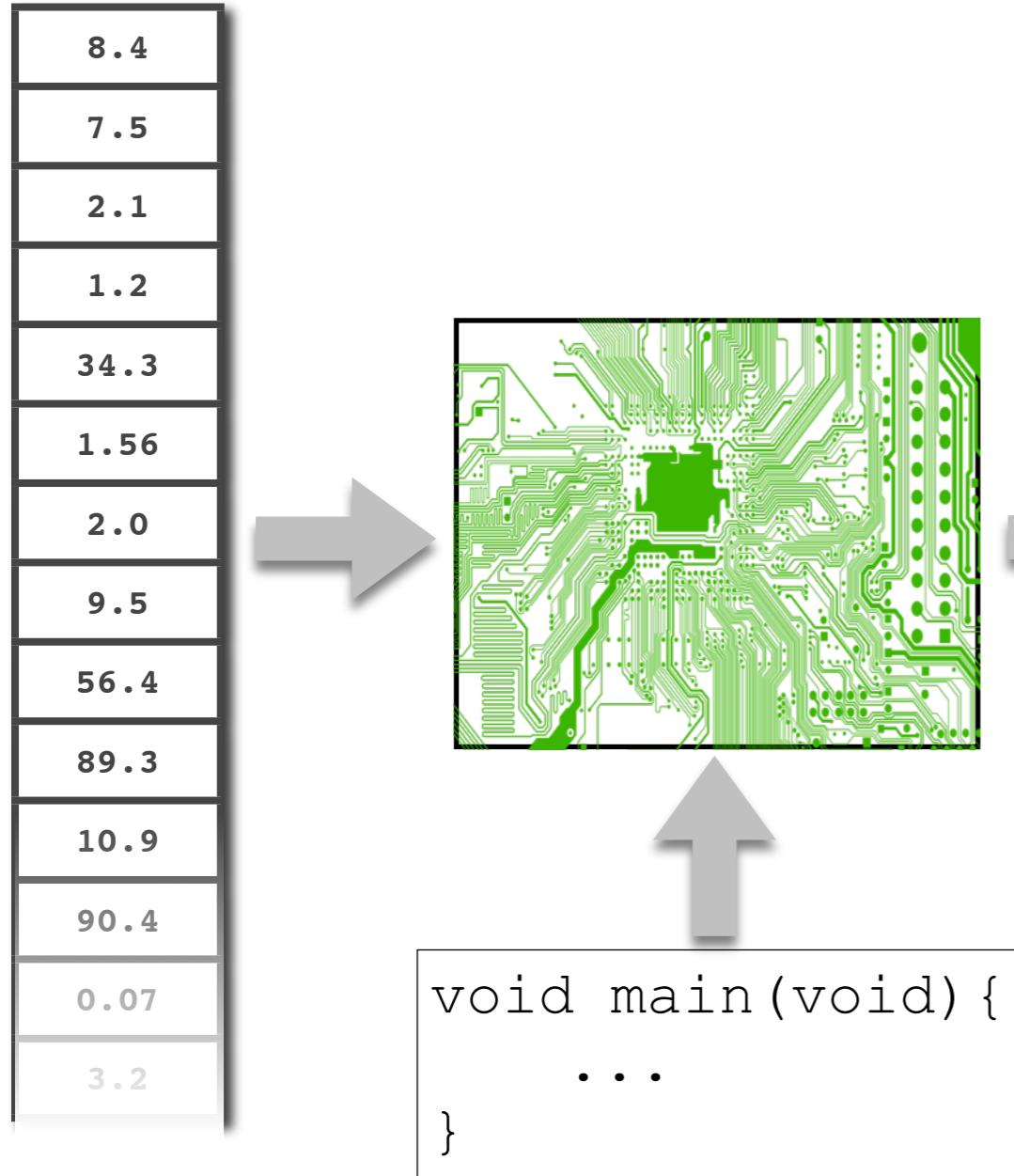
+

A 5x5 grid of numbers with colored highlights. The grid is composed of three nested rectangles:

- An innermost red rectangle containing the numbers 97, 8, 1, 8, 1; 43, 7, 34, 24, 3; 78, 45, 1; 3, 1, 95, 19, 21; 47, 3, 65, 6, 35; and 54, 2.
- A middle green rectangle covering the entire inner grid plus the numbers 98 and 2.
- An outermost blue rectangle covering the entire grid plus the numbers 90, 0, 8, 4, 8; 7, 12, 1, 43, 56; 64, 34, 1, 32, 89.

97	8	1	8	1
43	7	34	24	3
78	45	1	98	2
3	1	95	19	21
47	3	65	6	35
54	2	90	0	8
		7	12	1
		64	34	1
		32	89	

# CGI pipeline

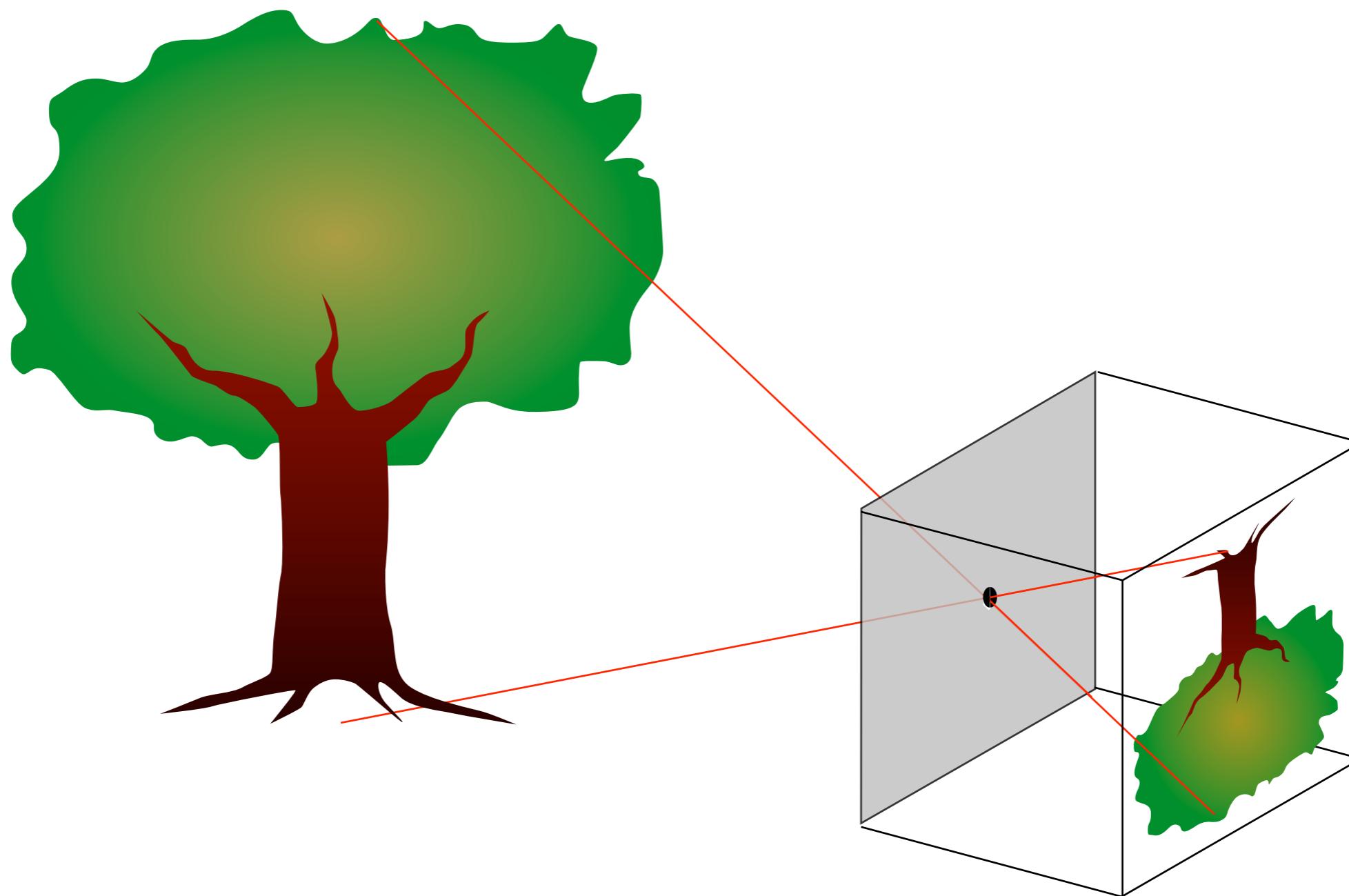


Représentation ✓

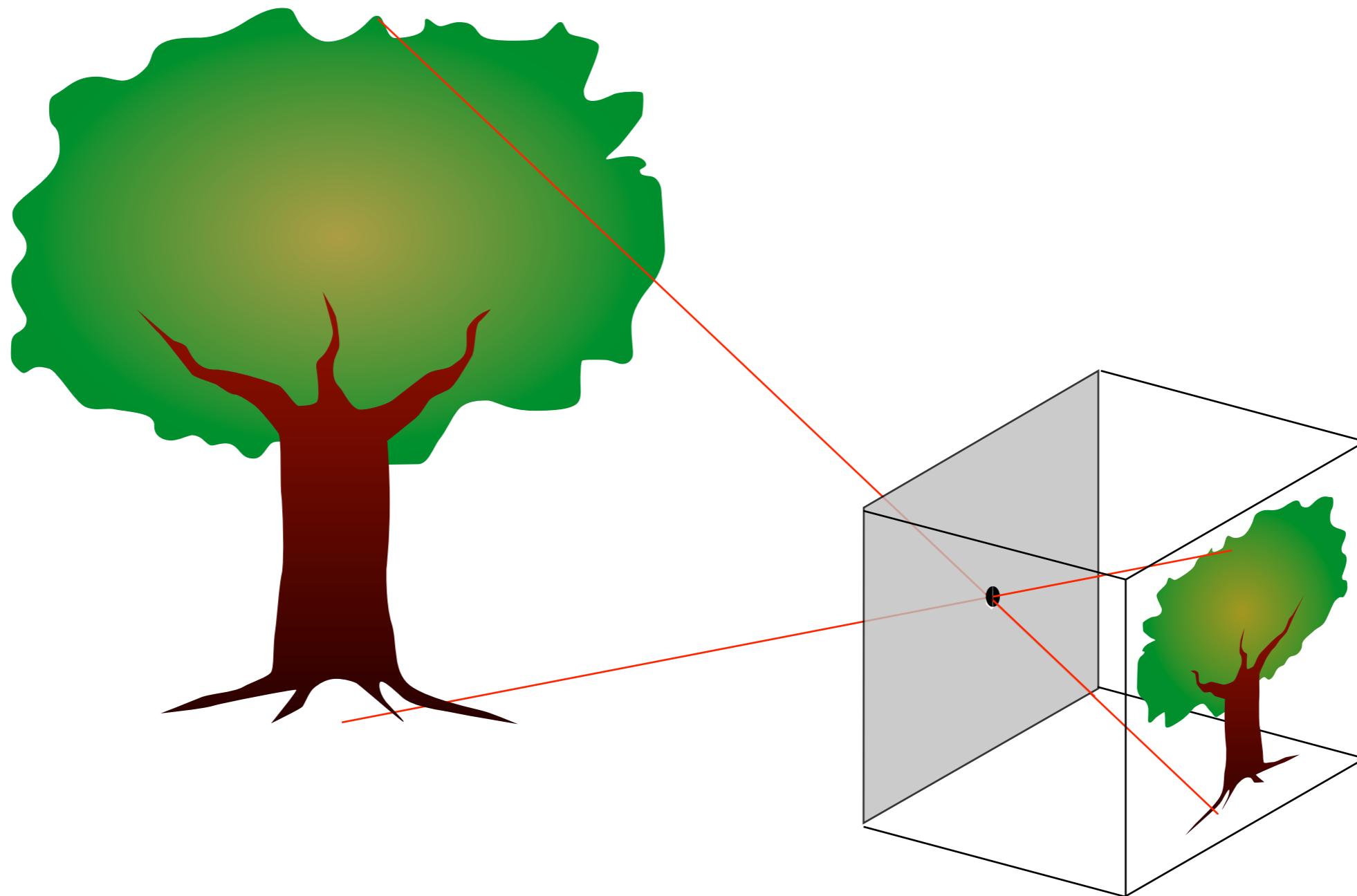
Algorithme ✗

Frame(s)

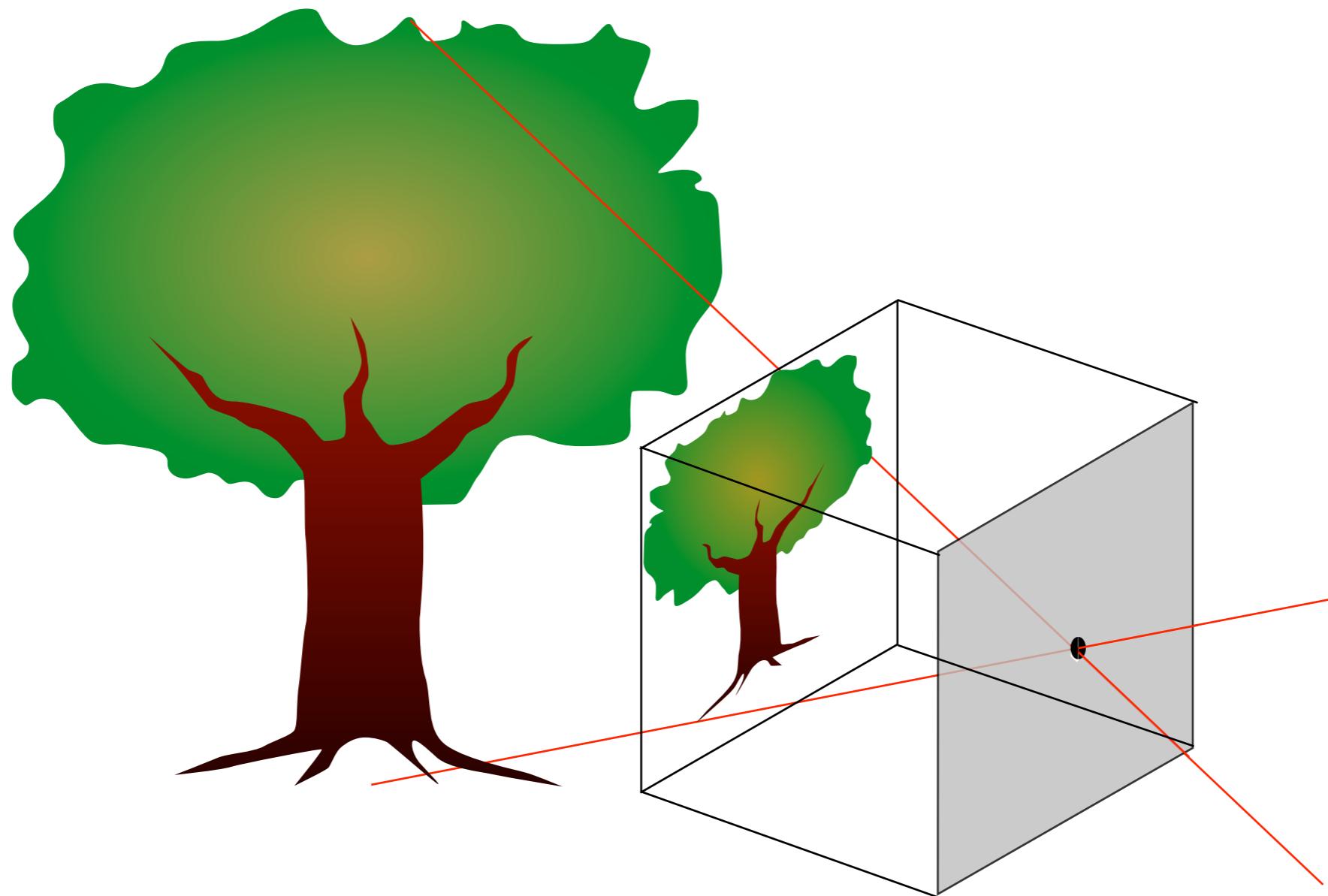
# Camera obscura



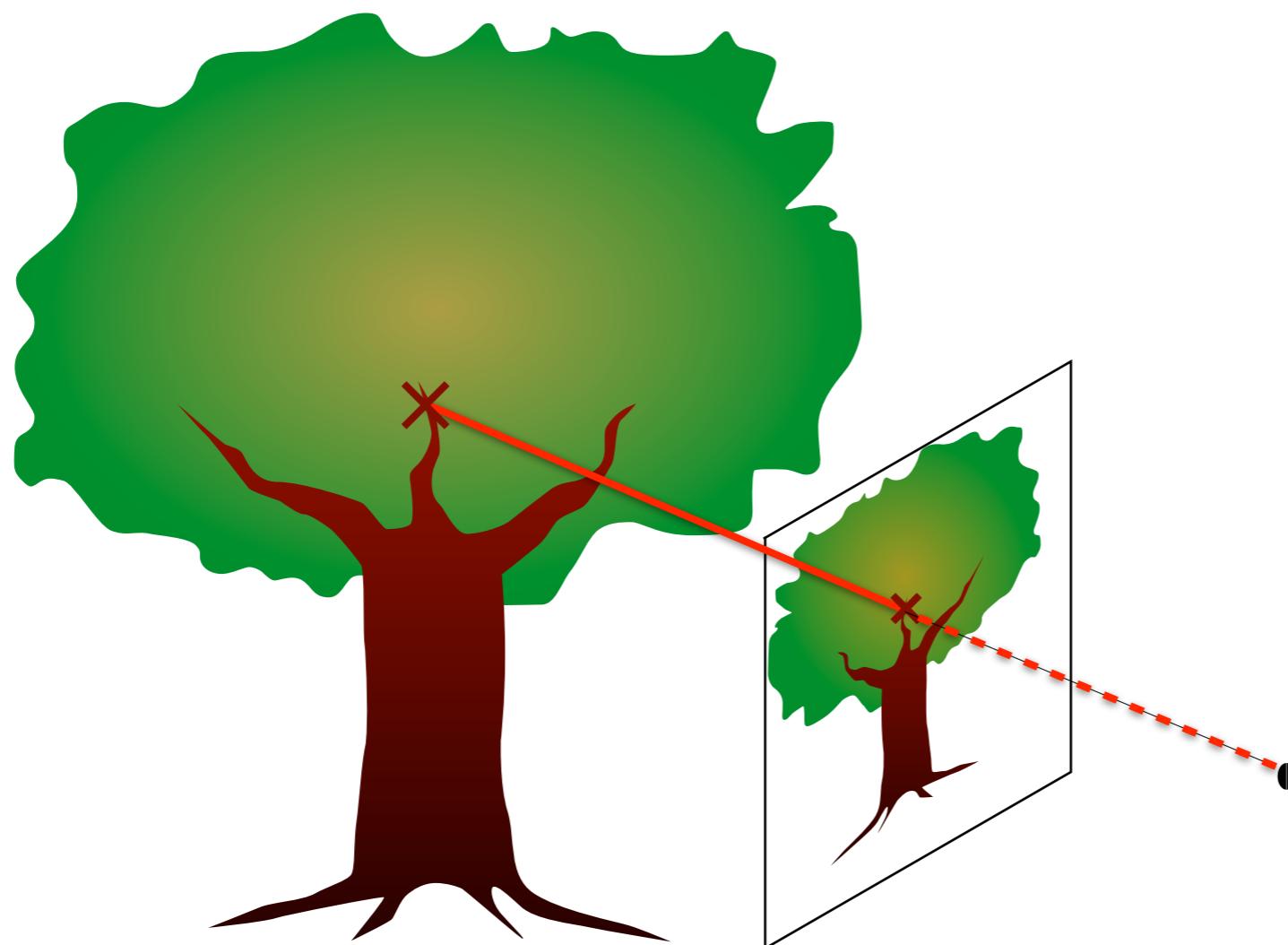
# Camera obscura



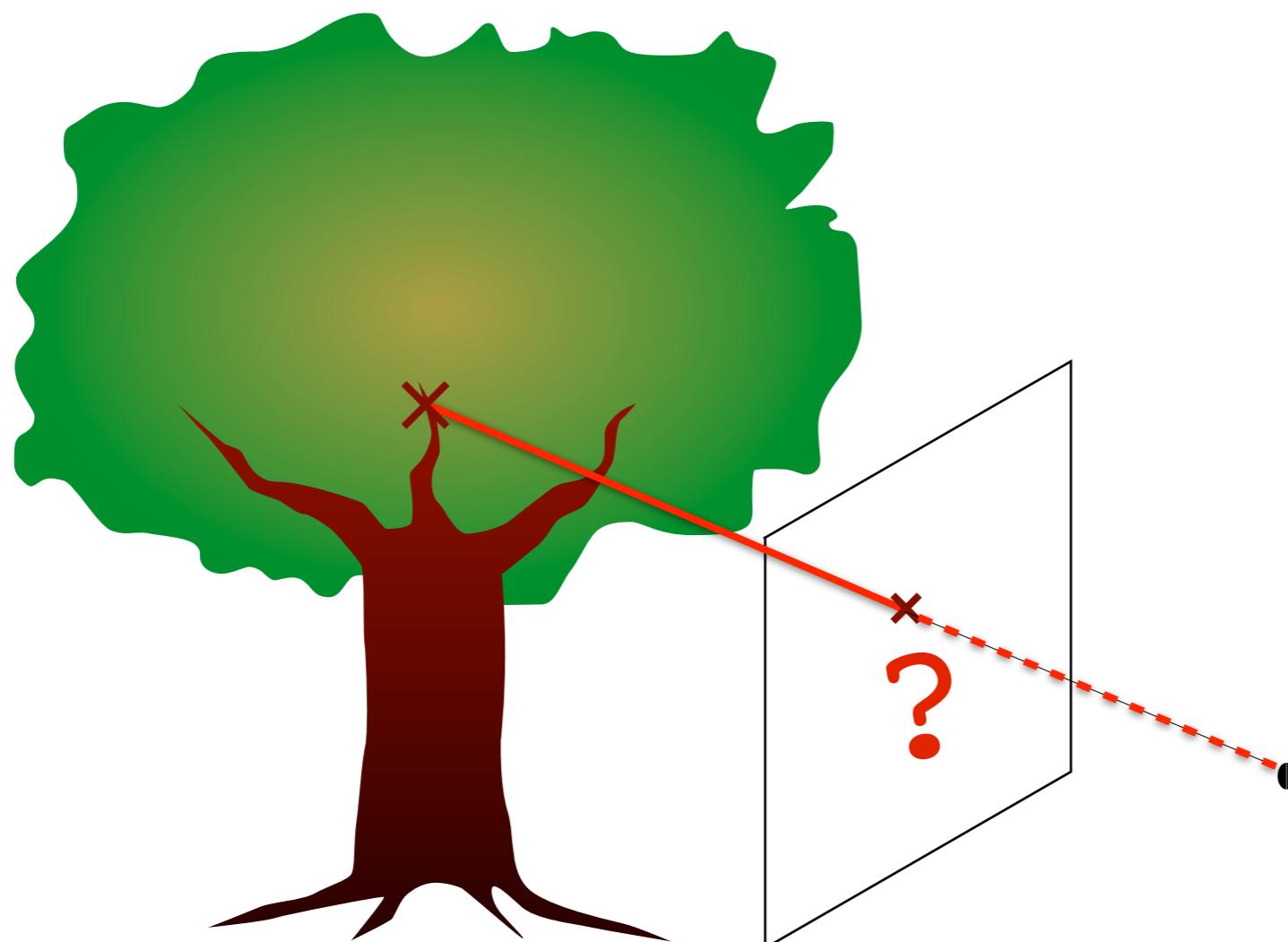
# Camera sténopée



# Camera sténopée

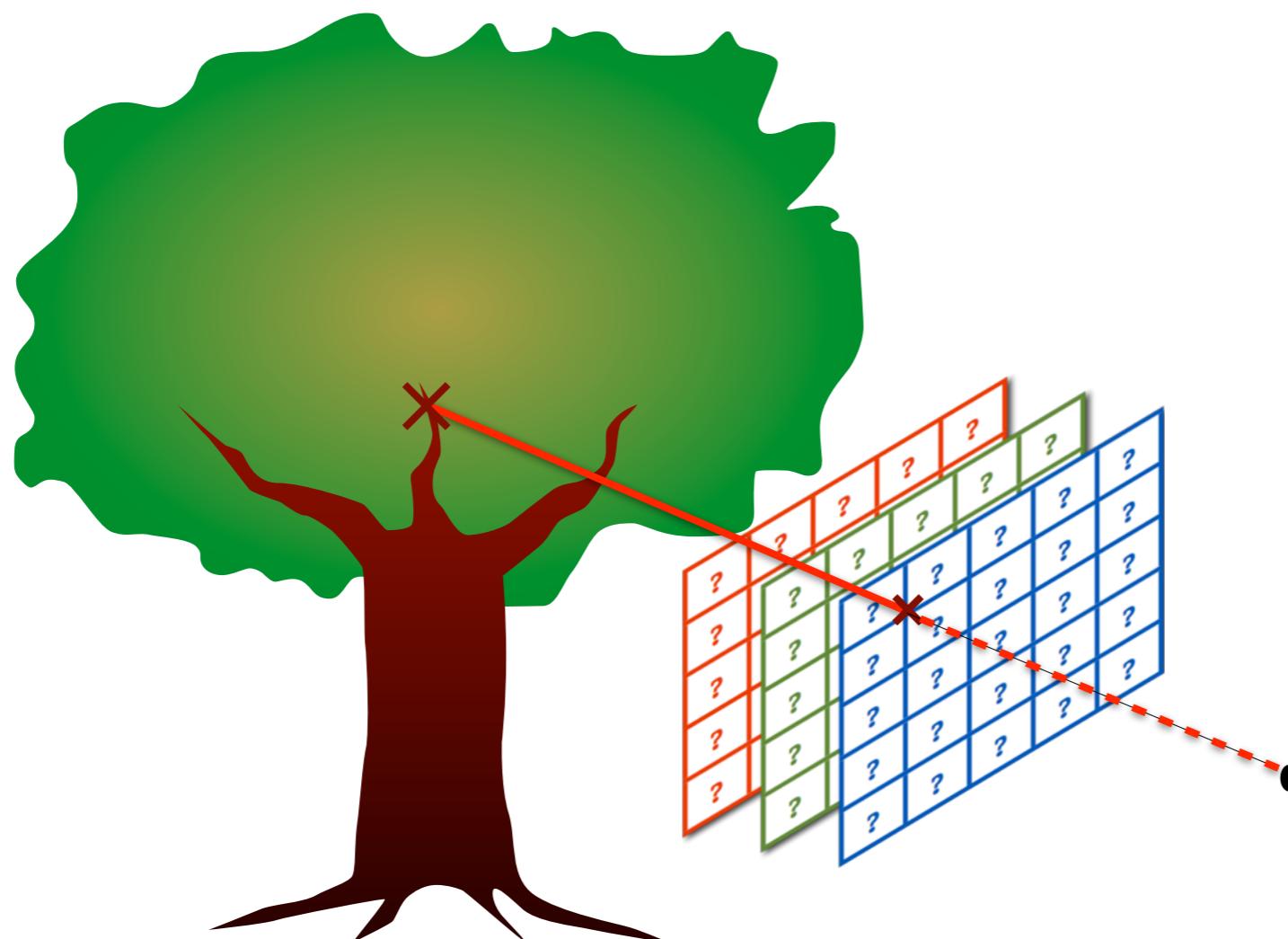


# Camera sténopée



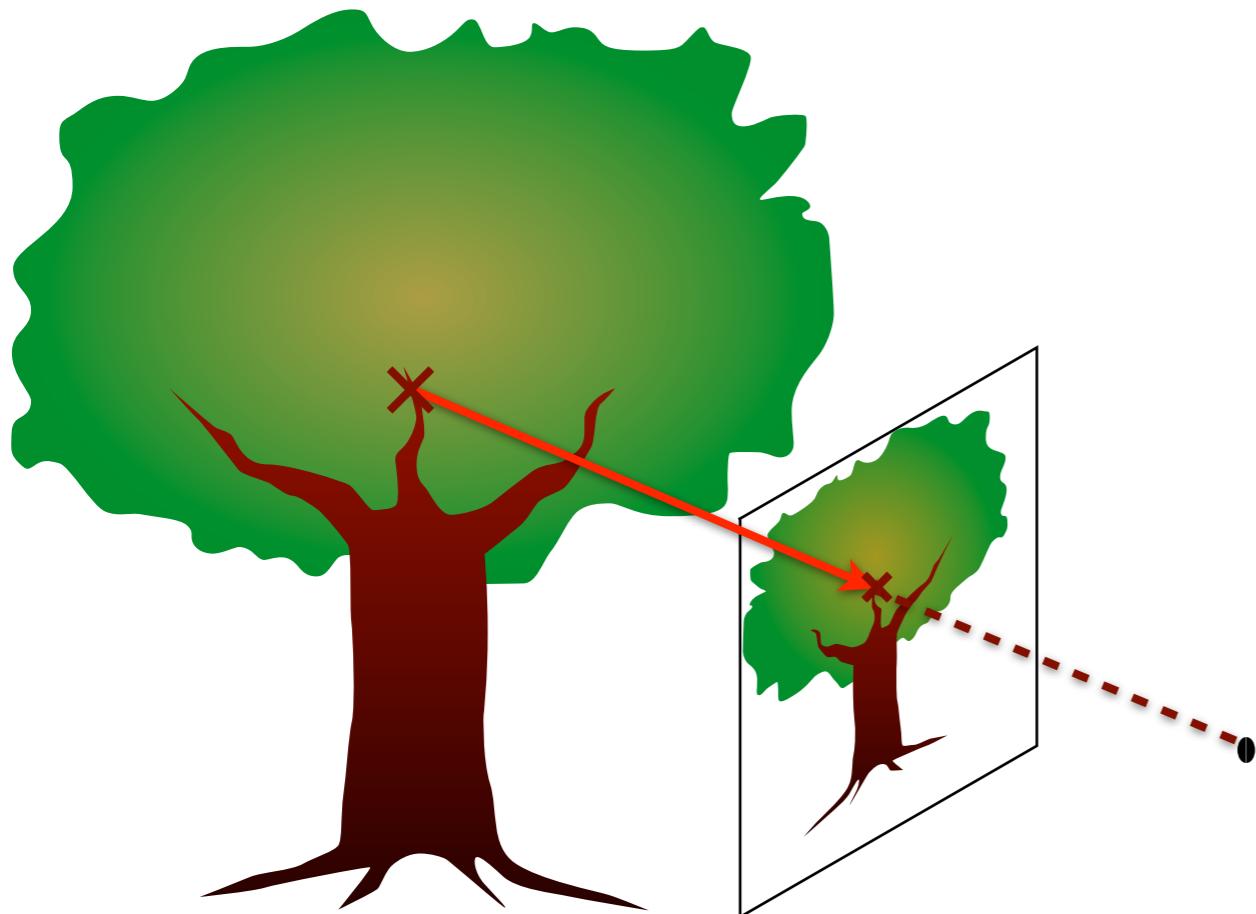
Simulation

# Camera sténopée

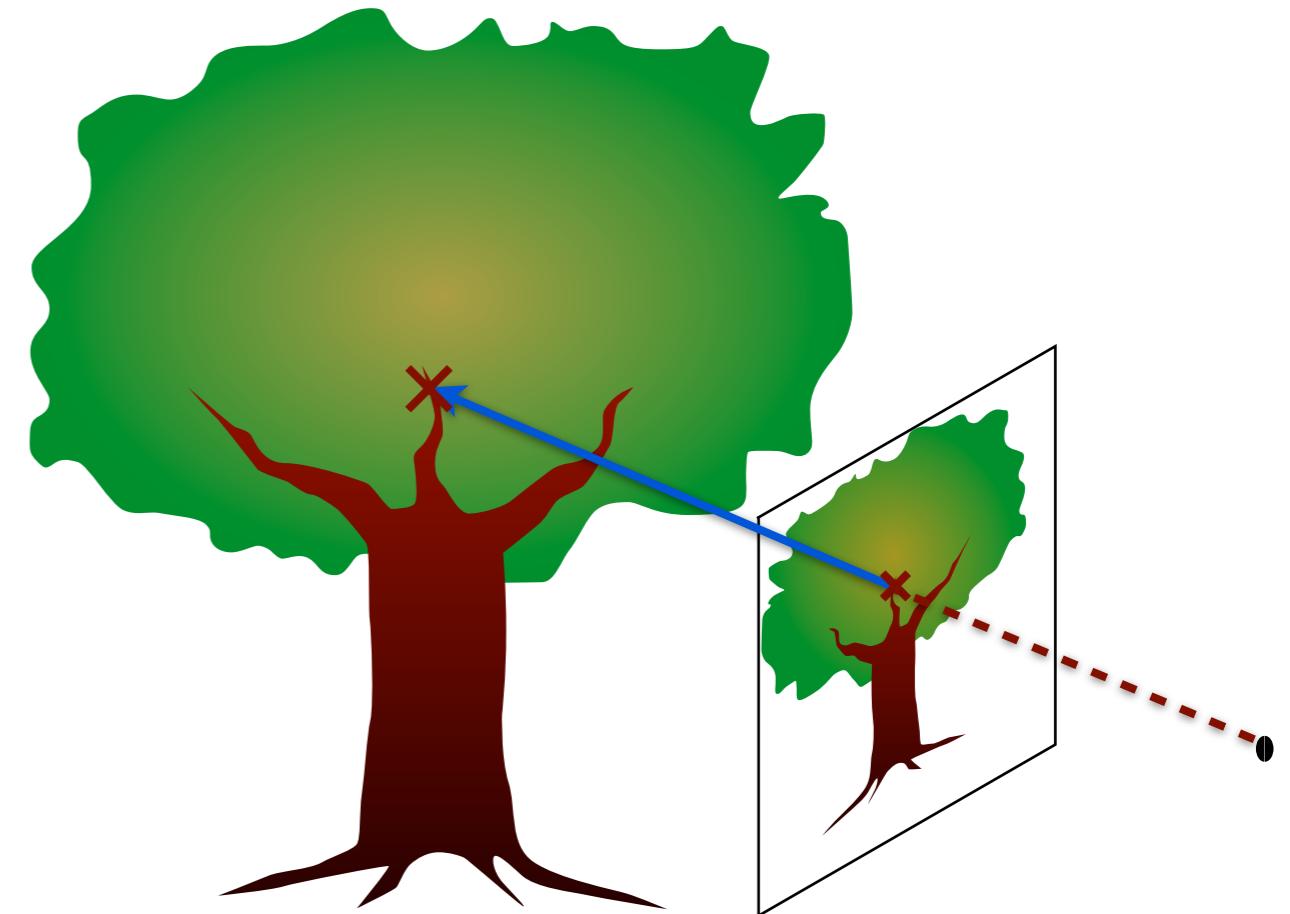


Simulation

# Rasterization / Raytracing



**3D → 2D**

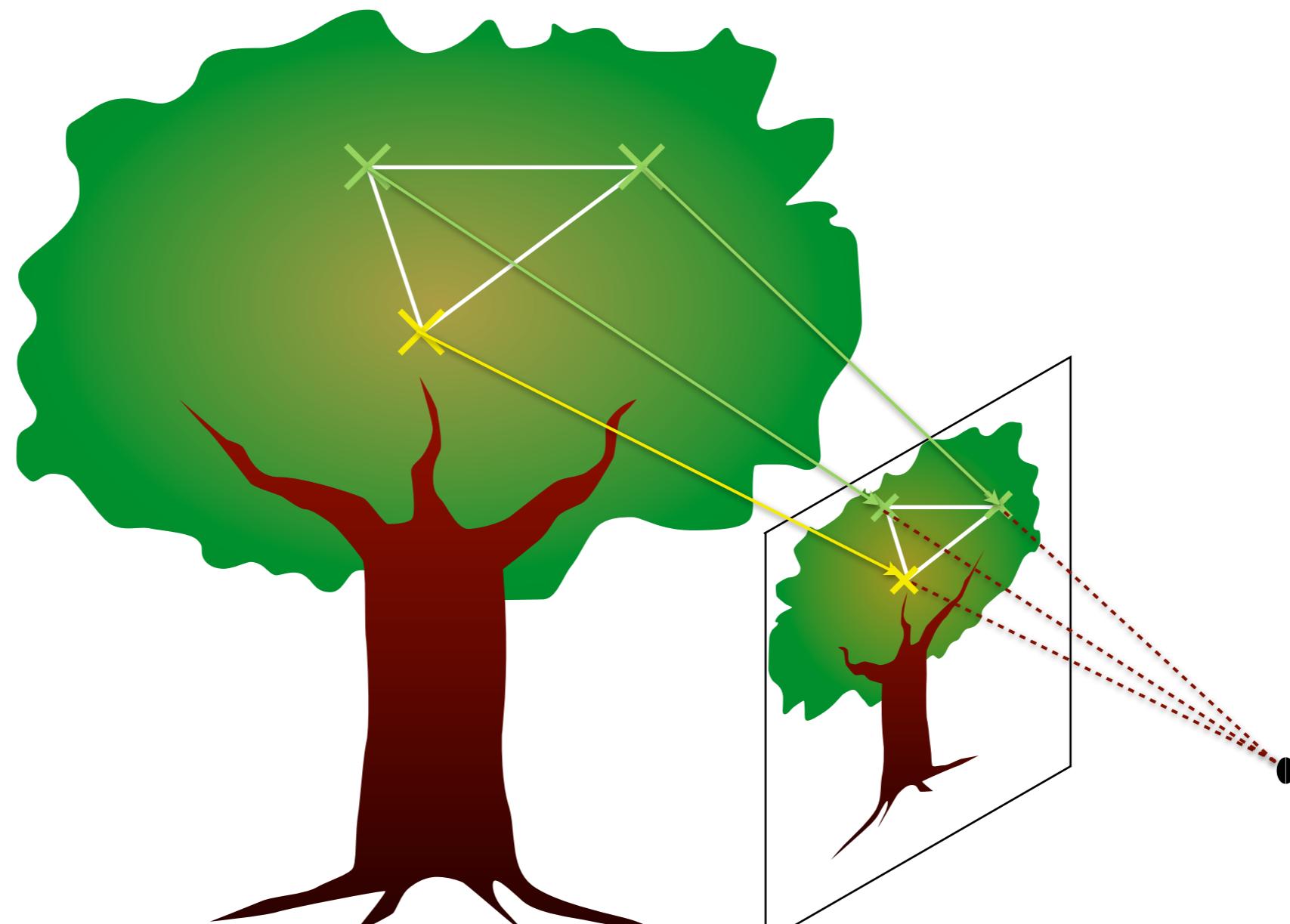


**2D → 3D**

# Algorithmes de CGI

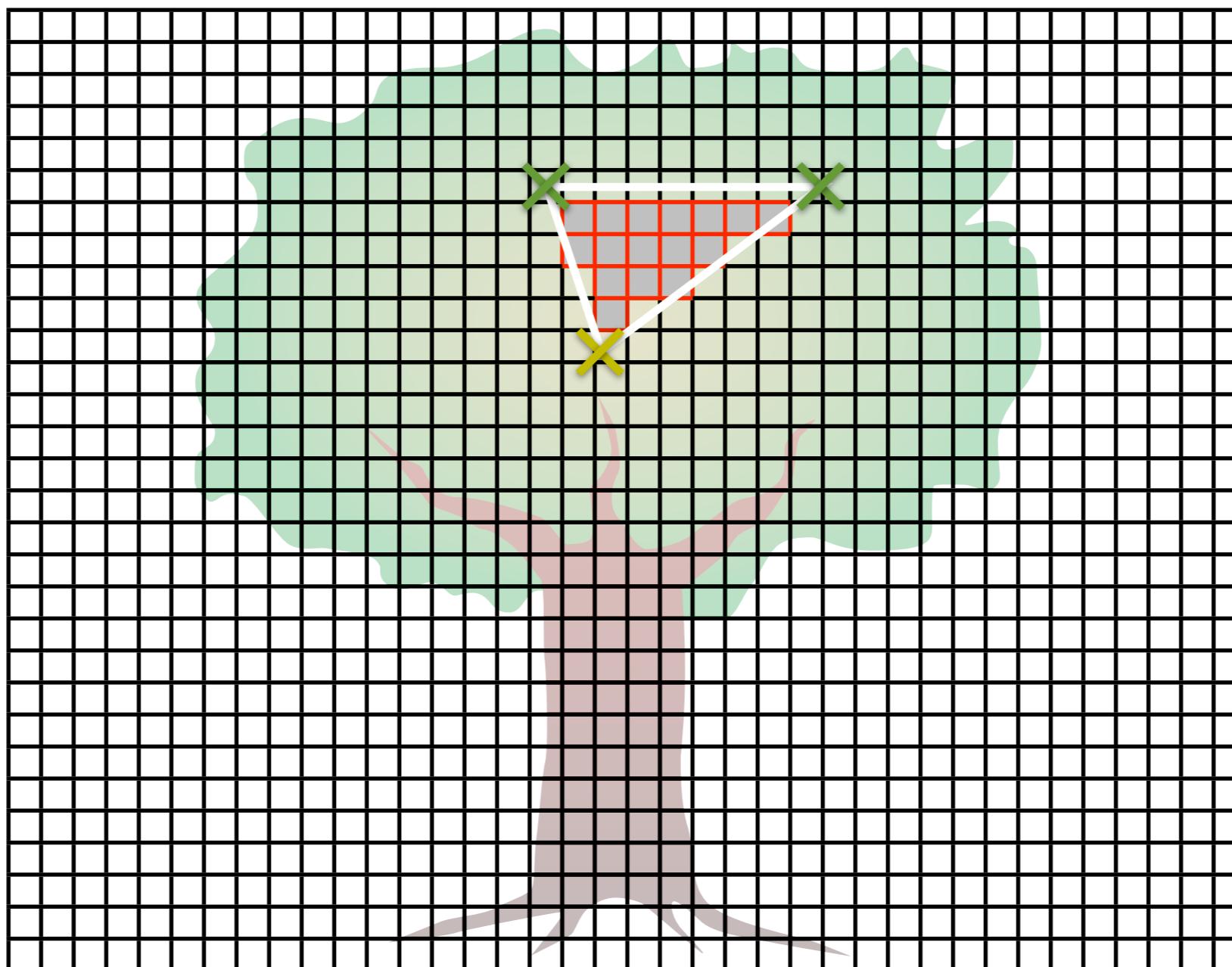
## Rasterization

# Rasterization



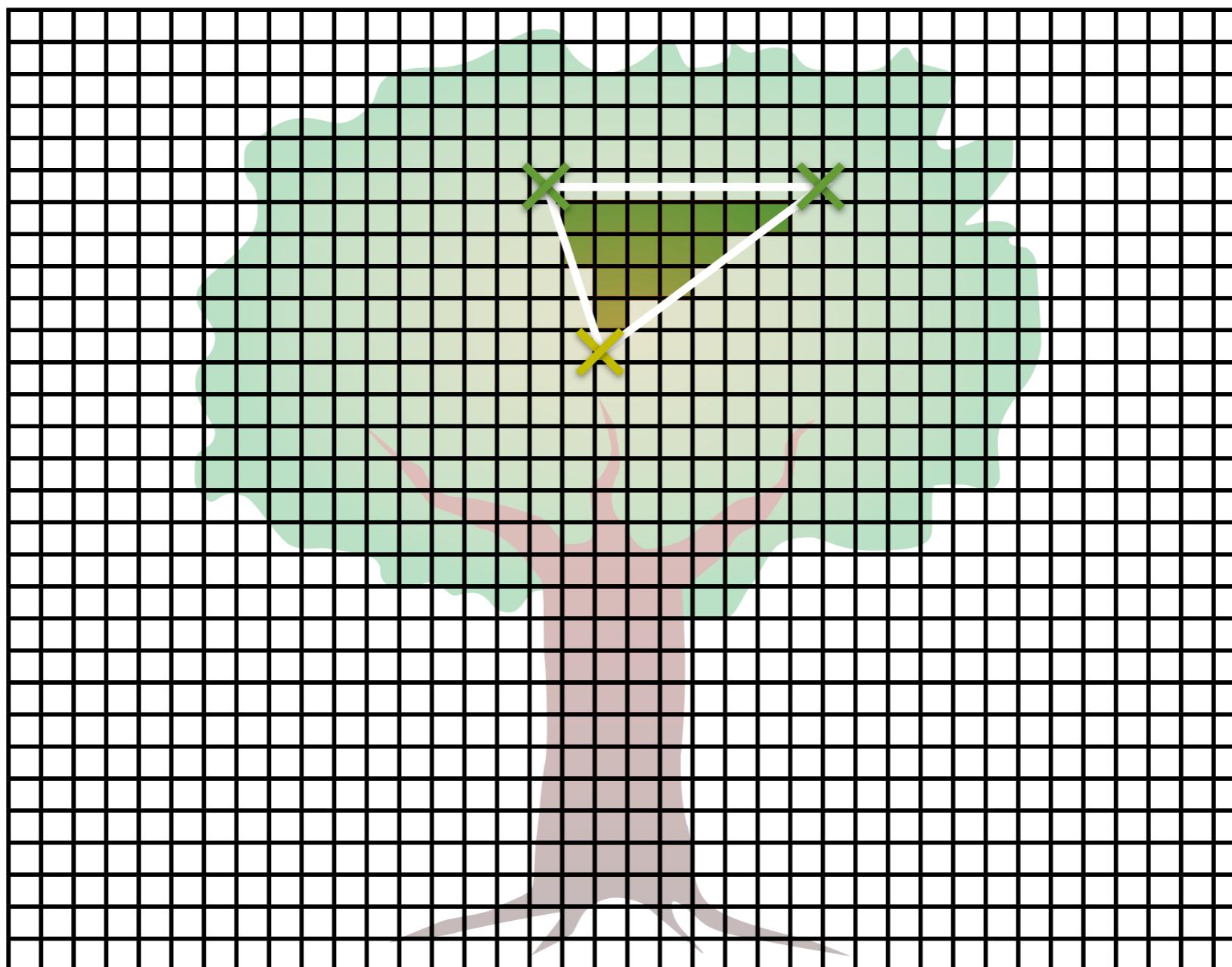
**3D → 2D**

# Rasterization



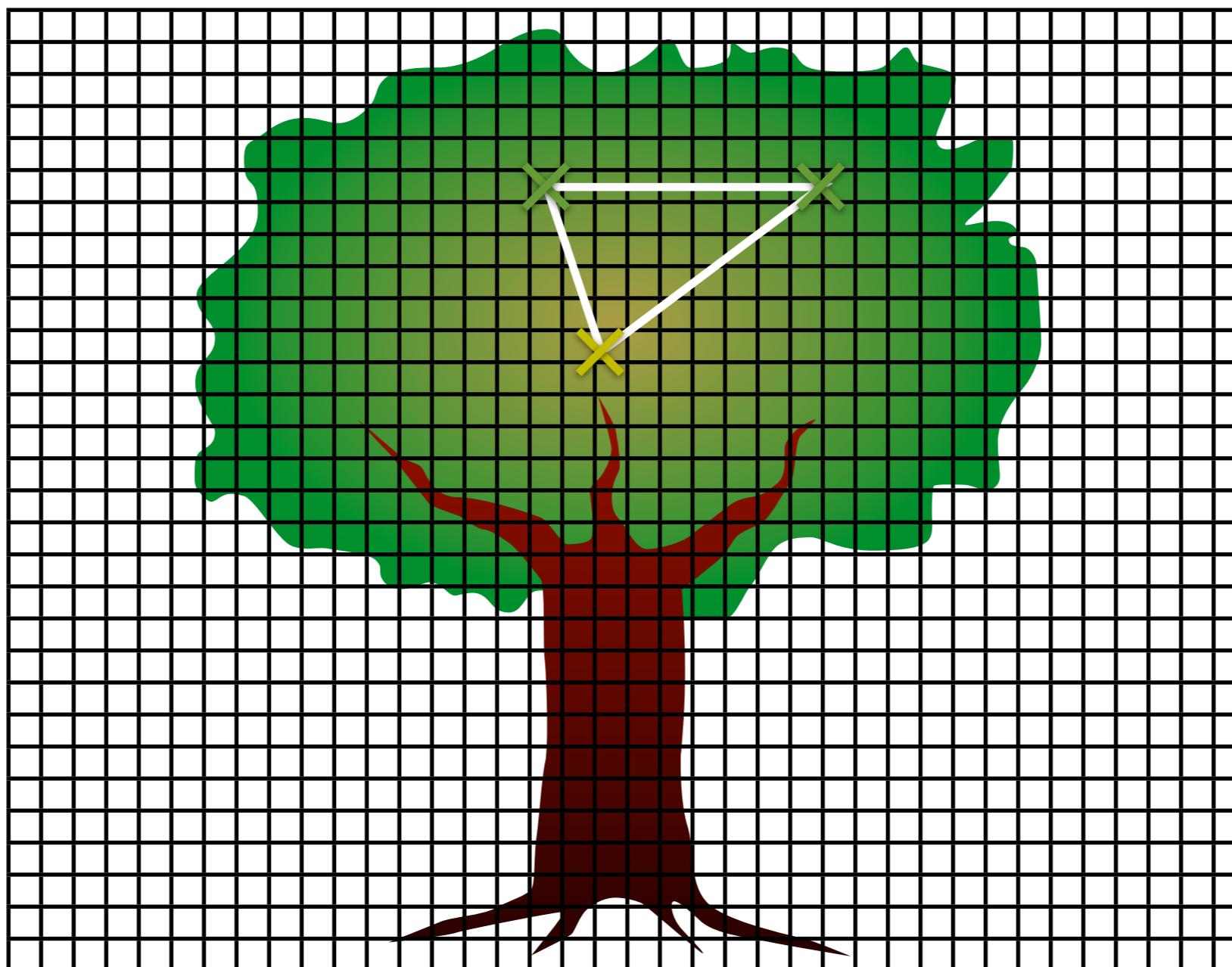
**3D → 2D**

# Rasterization



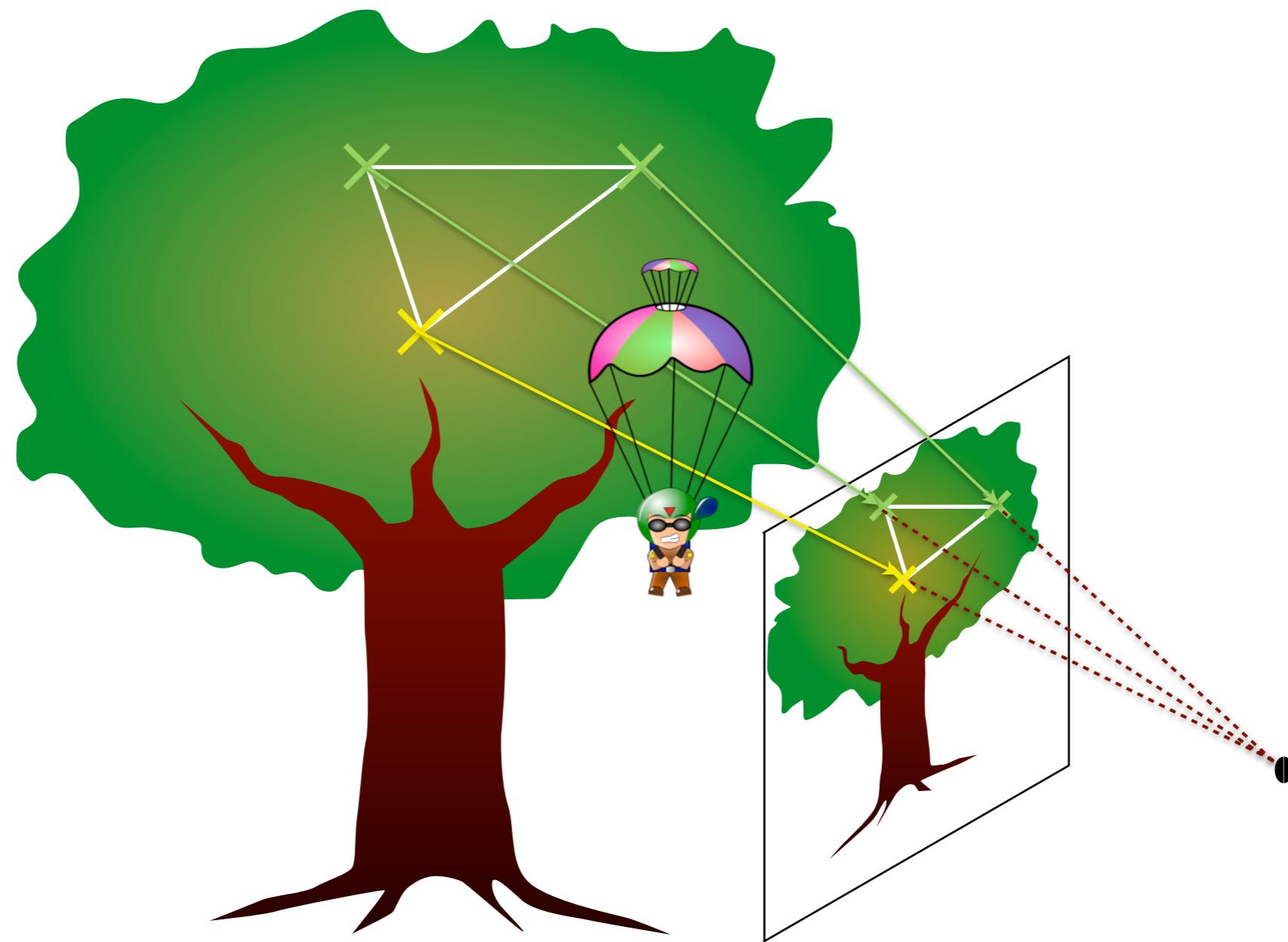
**3D → 2D**

# Rasterization



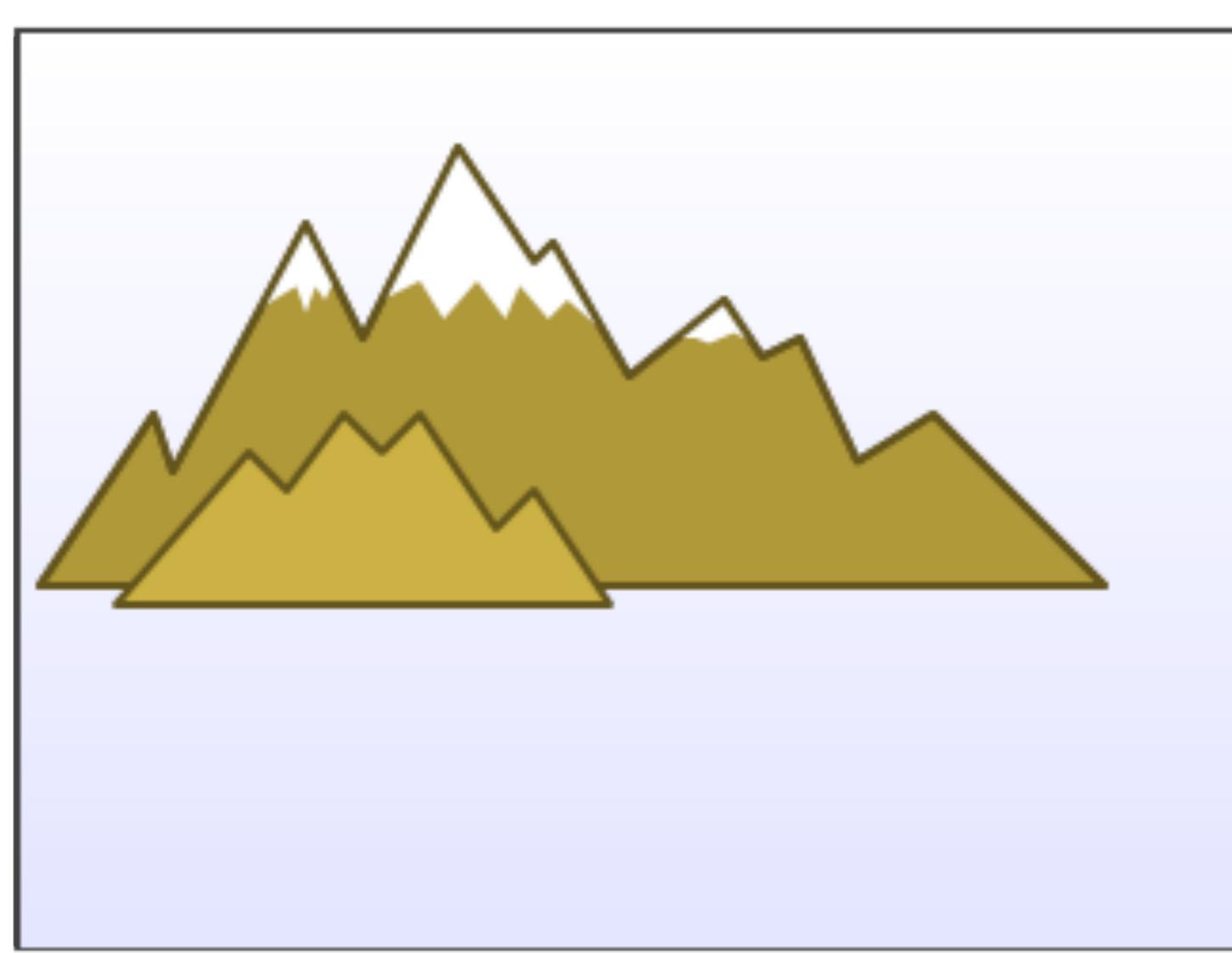
3D → 2D

# Occlusions

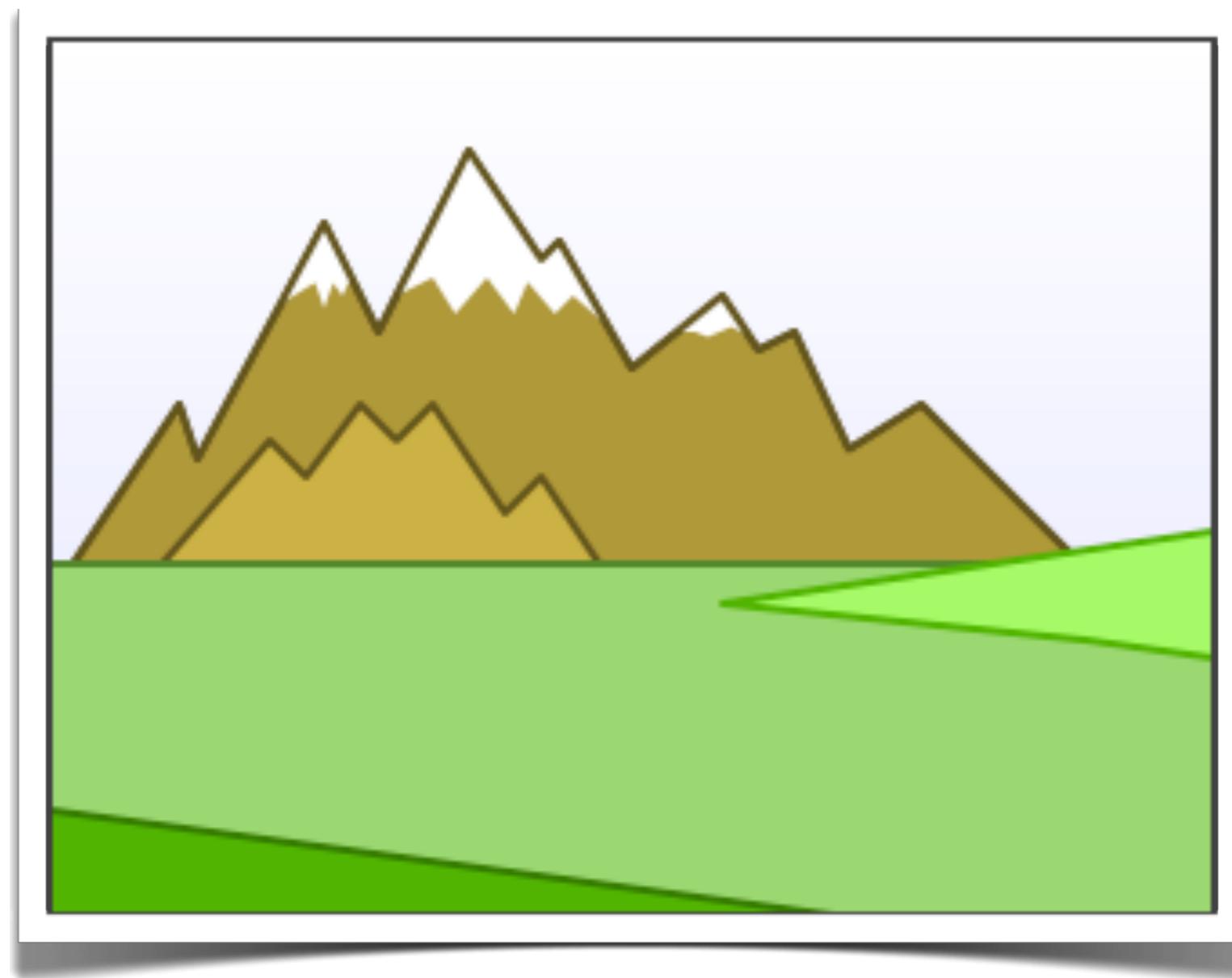


Ordre compte

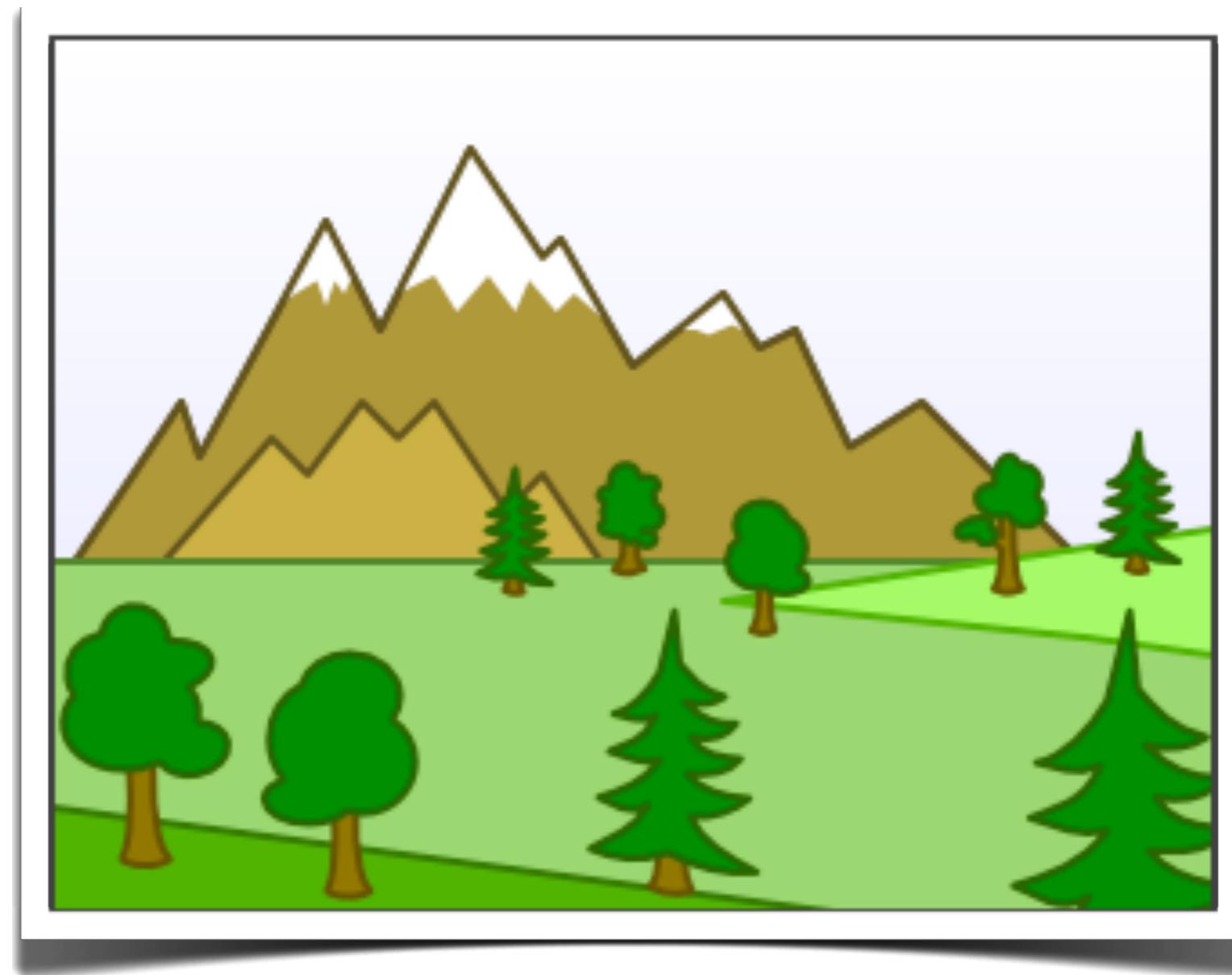
# Algorithme du peintre



# Algorithme du peintre

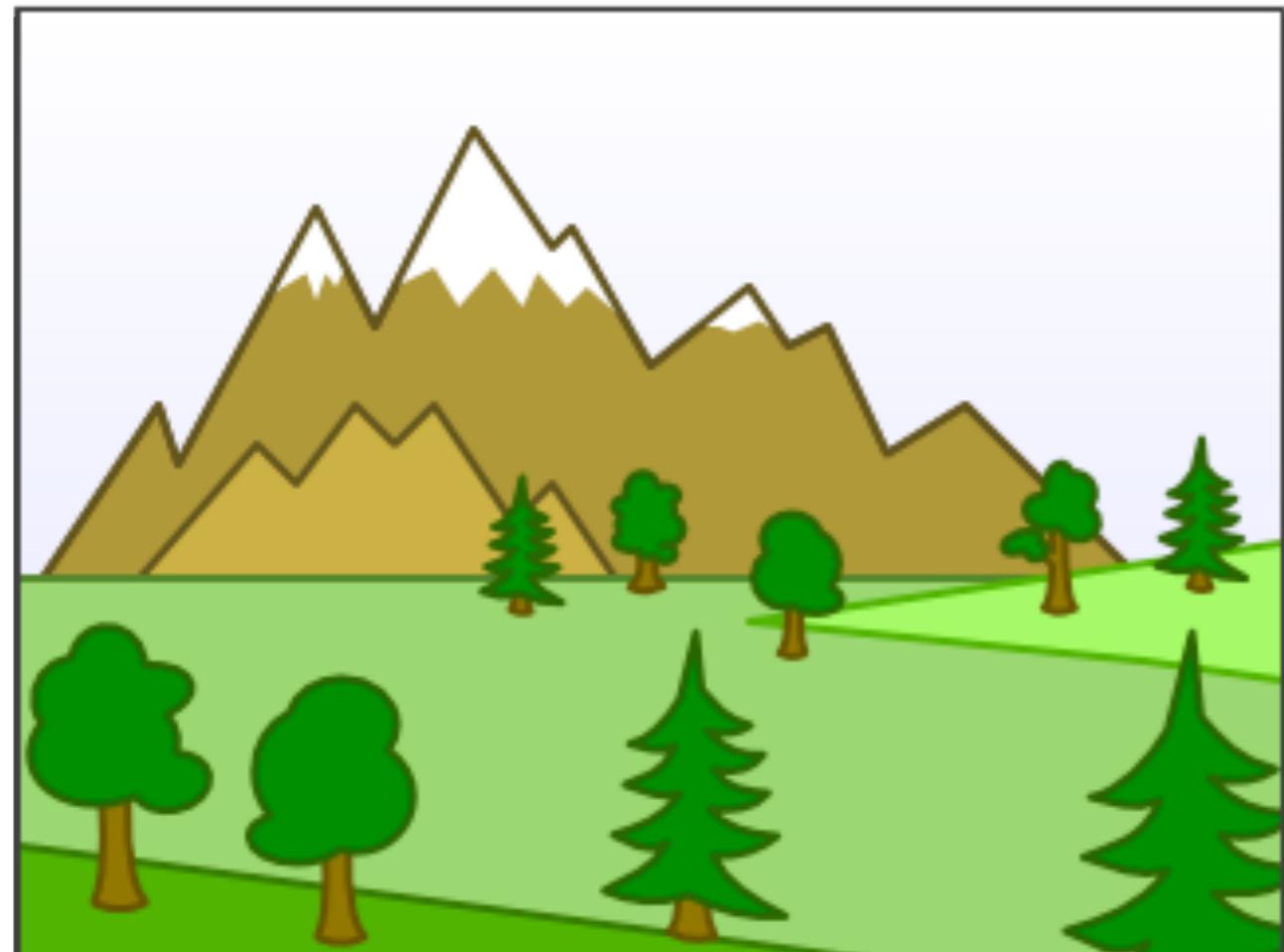


# Algorithme du peintre

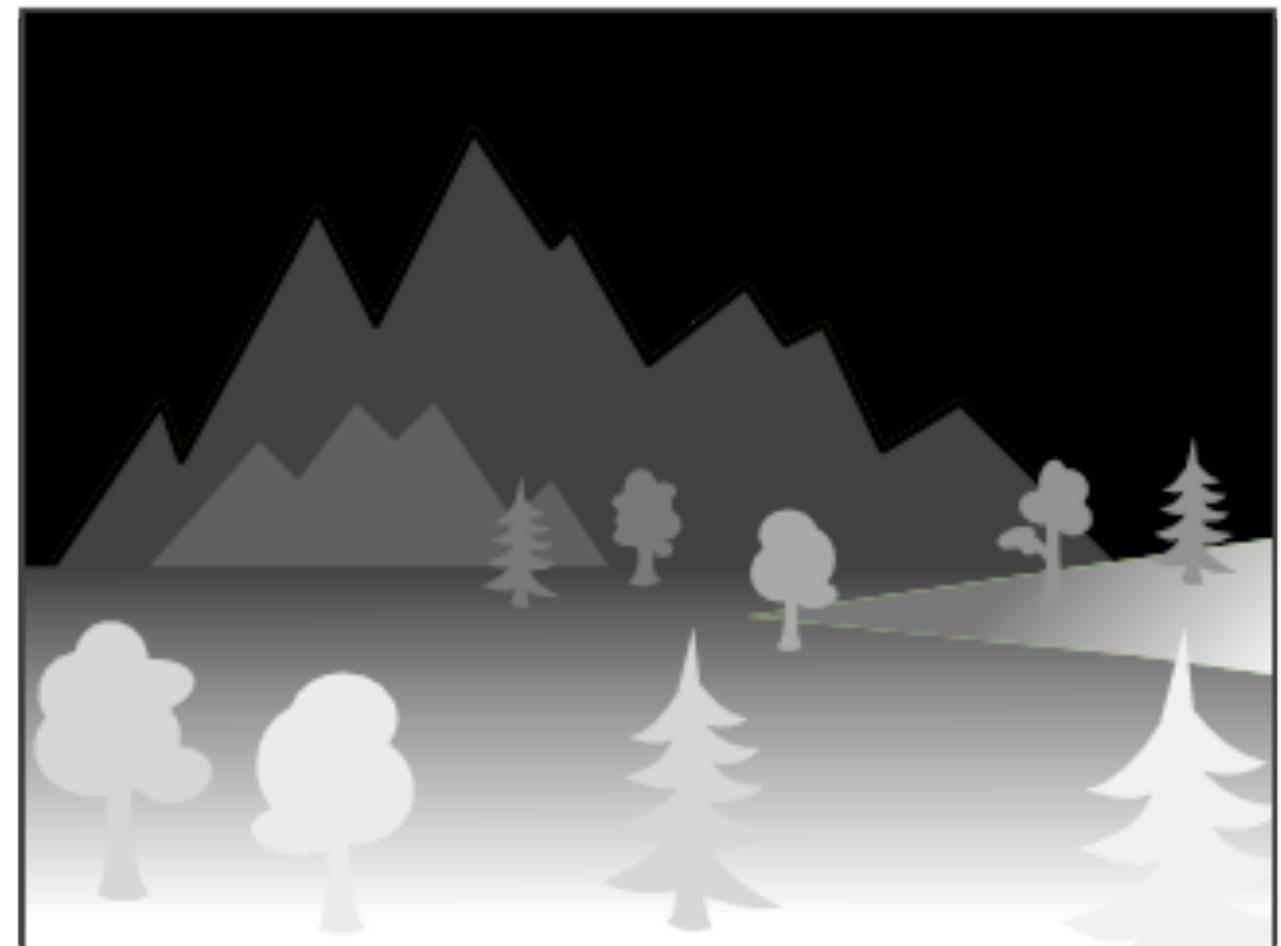


Compliqué

# Z-buffer (Catmull)

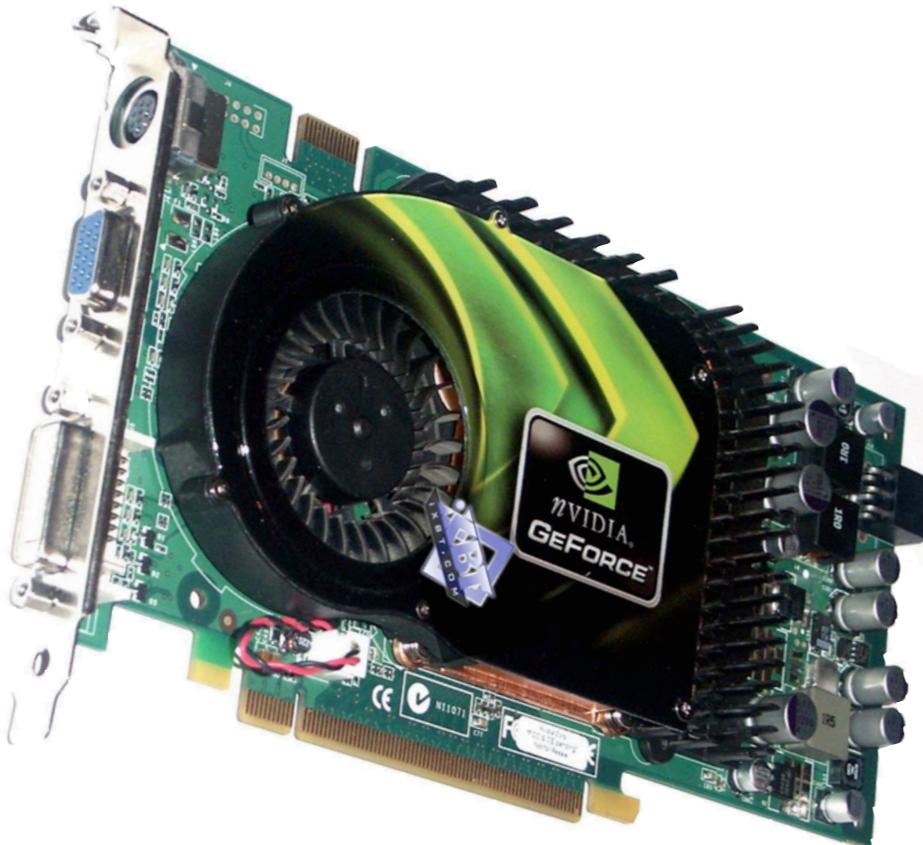


Color map

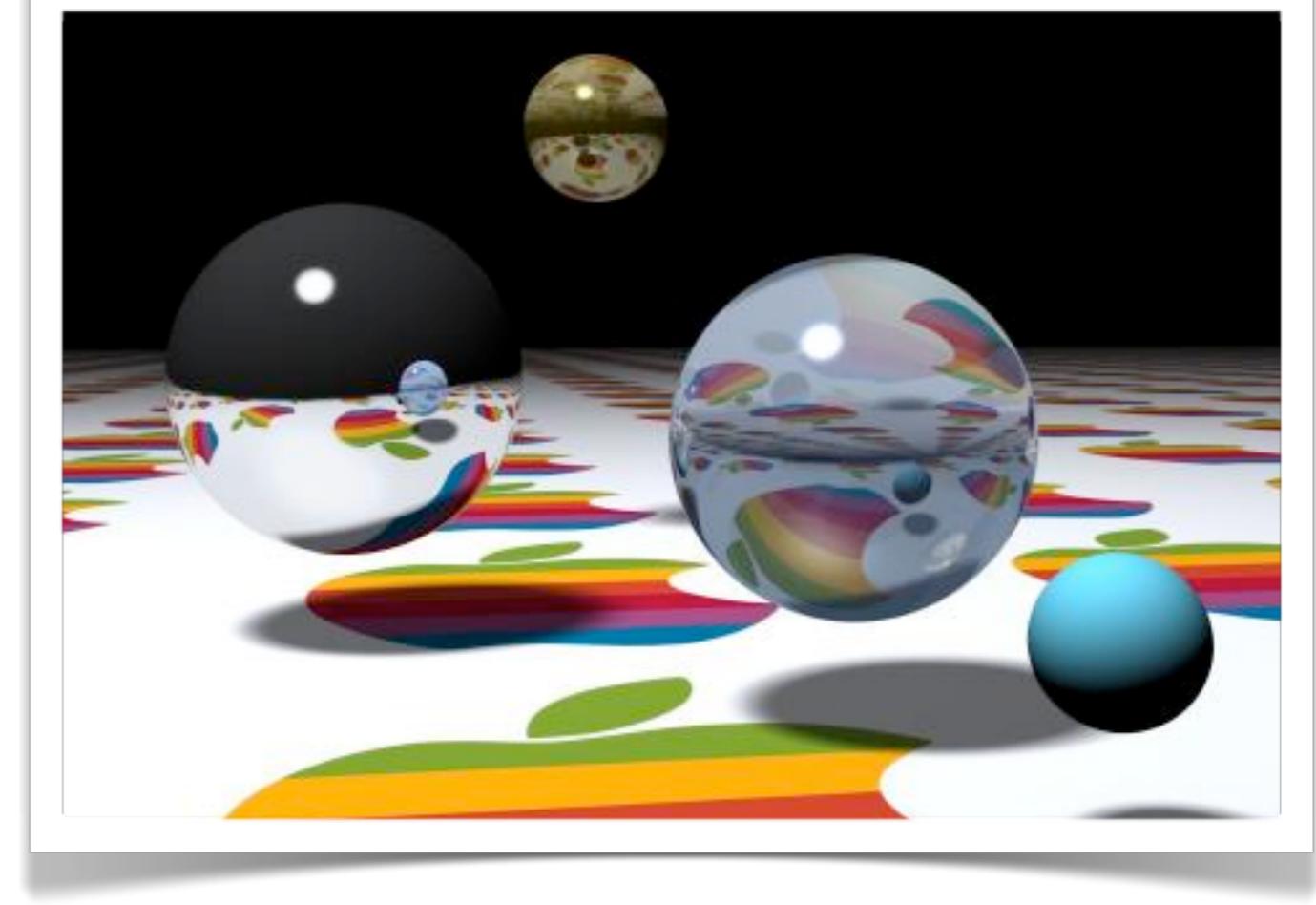


Depth map

# Pros & cons



Fast implementation (GPU)

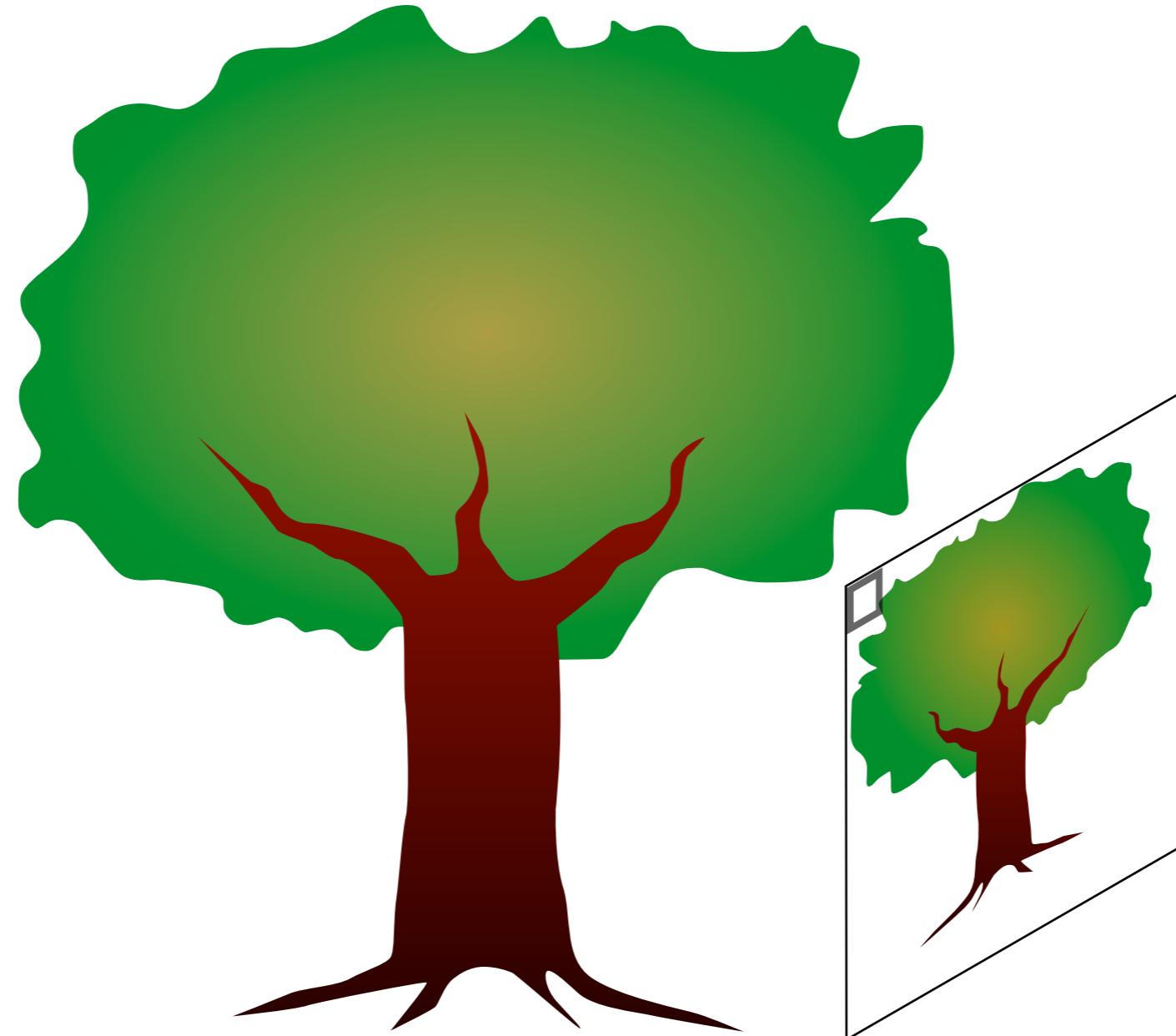


Shadows, reflexions,...

# Algorithmes de CGI

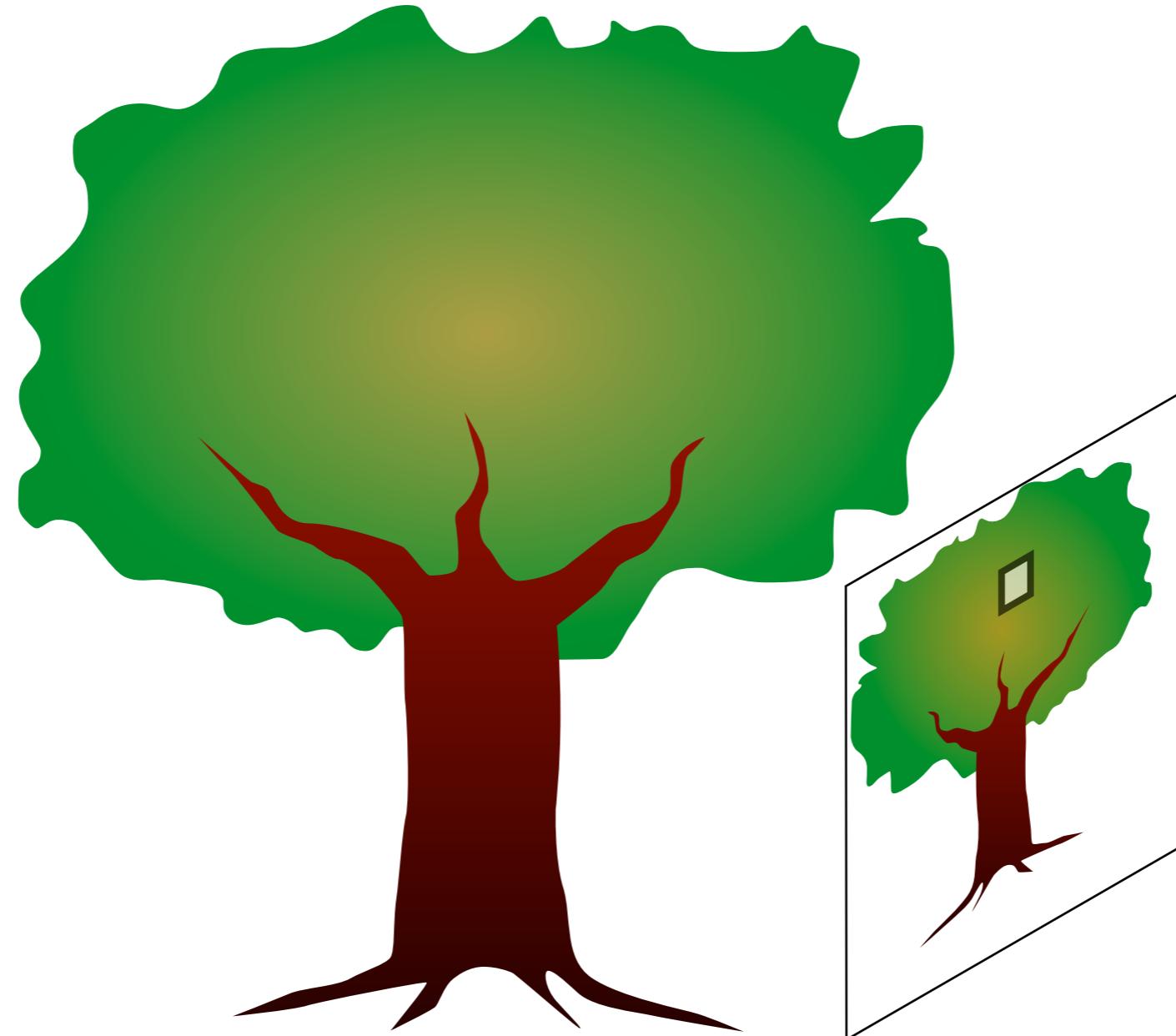
## Ray tracing

# Raytracing



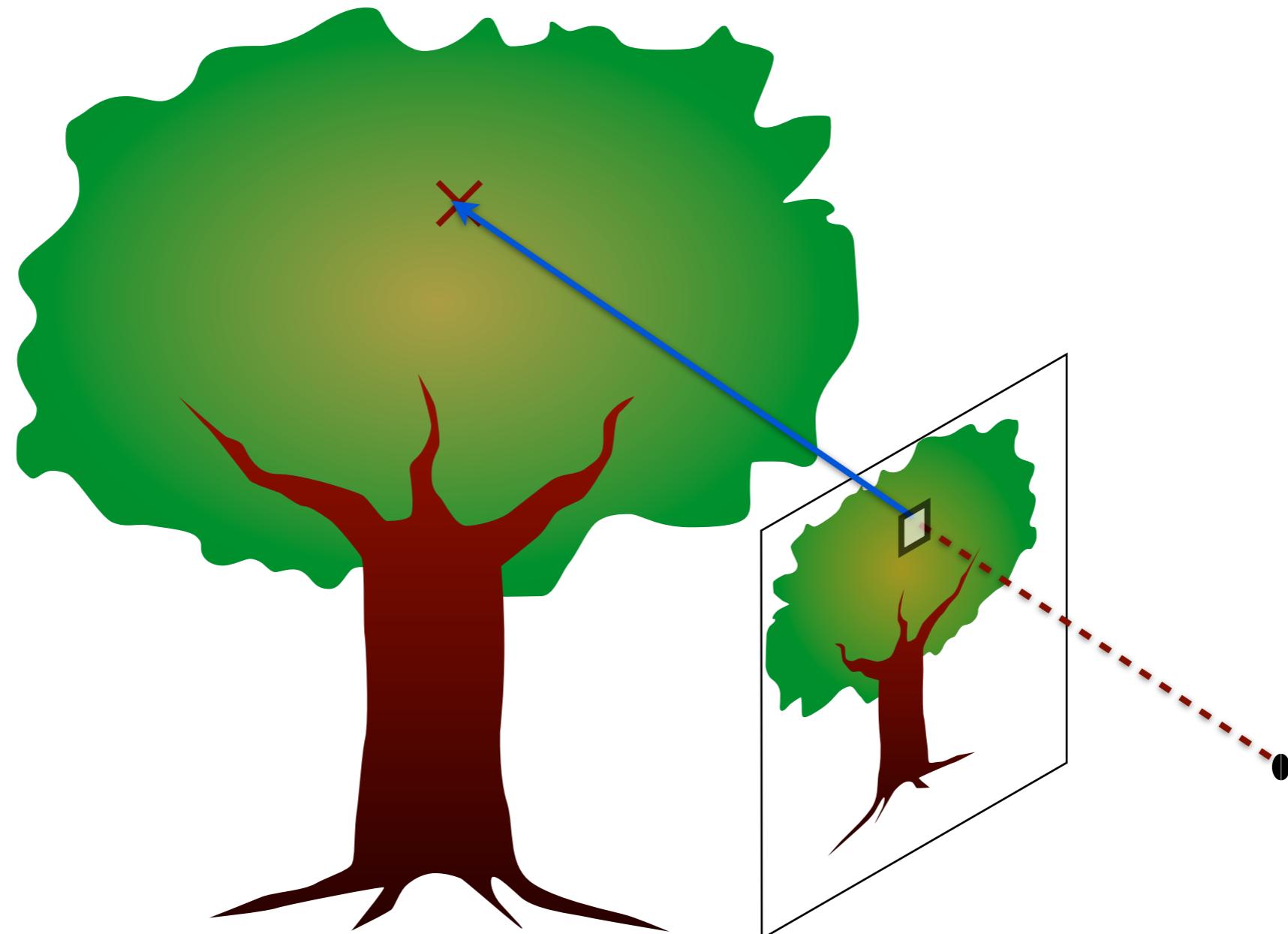
2D → 3D

# Raytracing



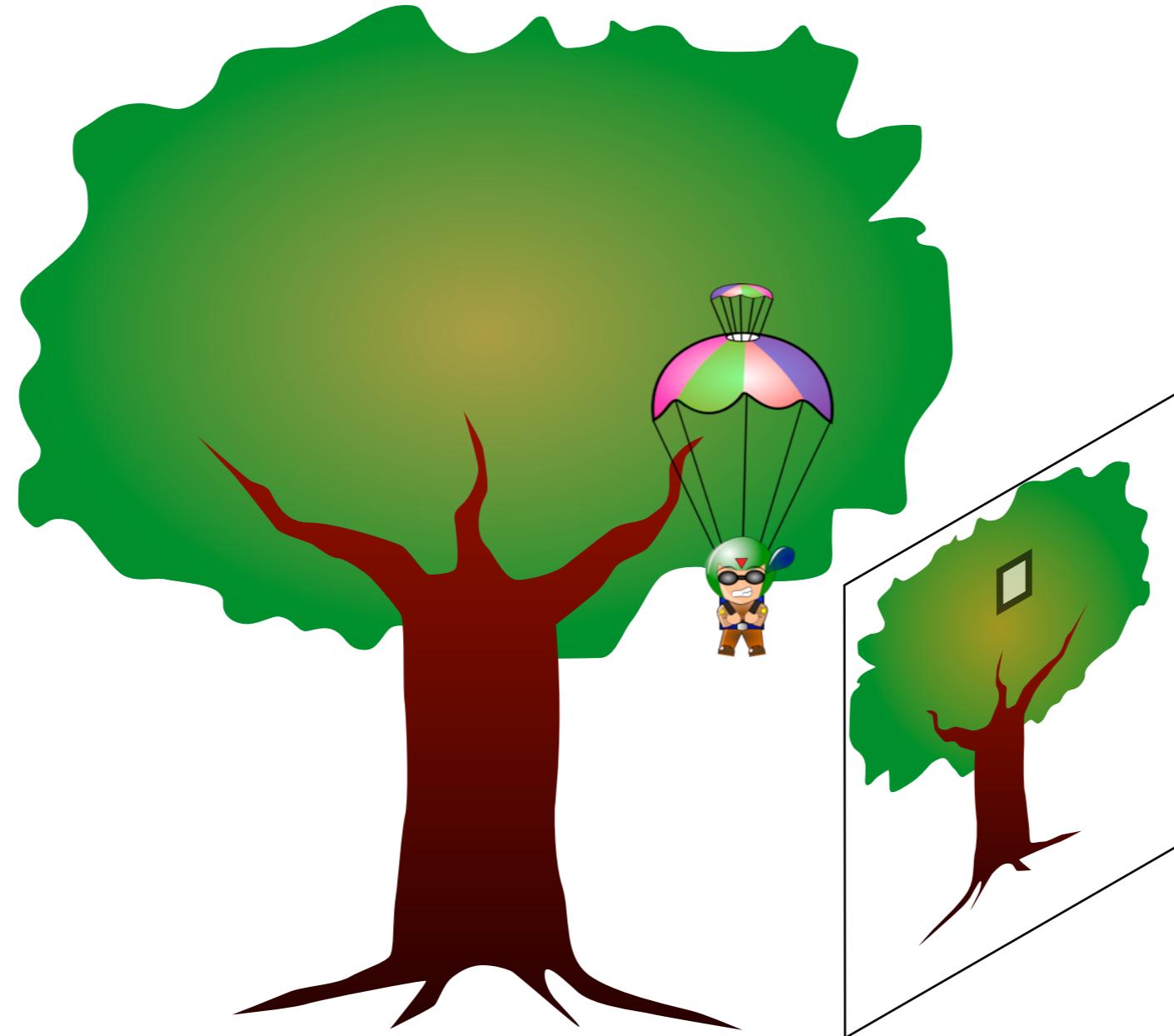
2D → 3D

# Raytracing



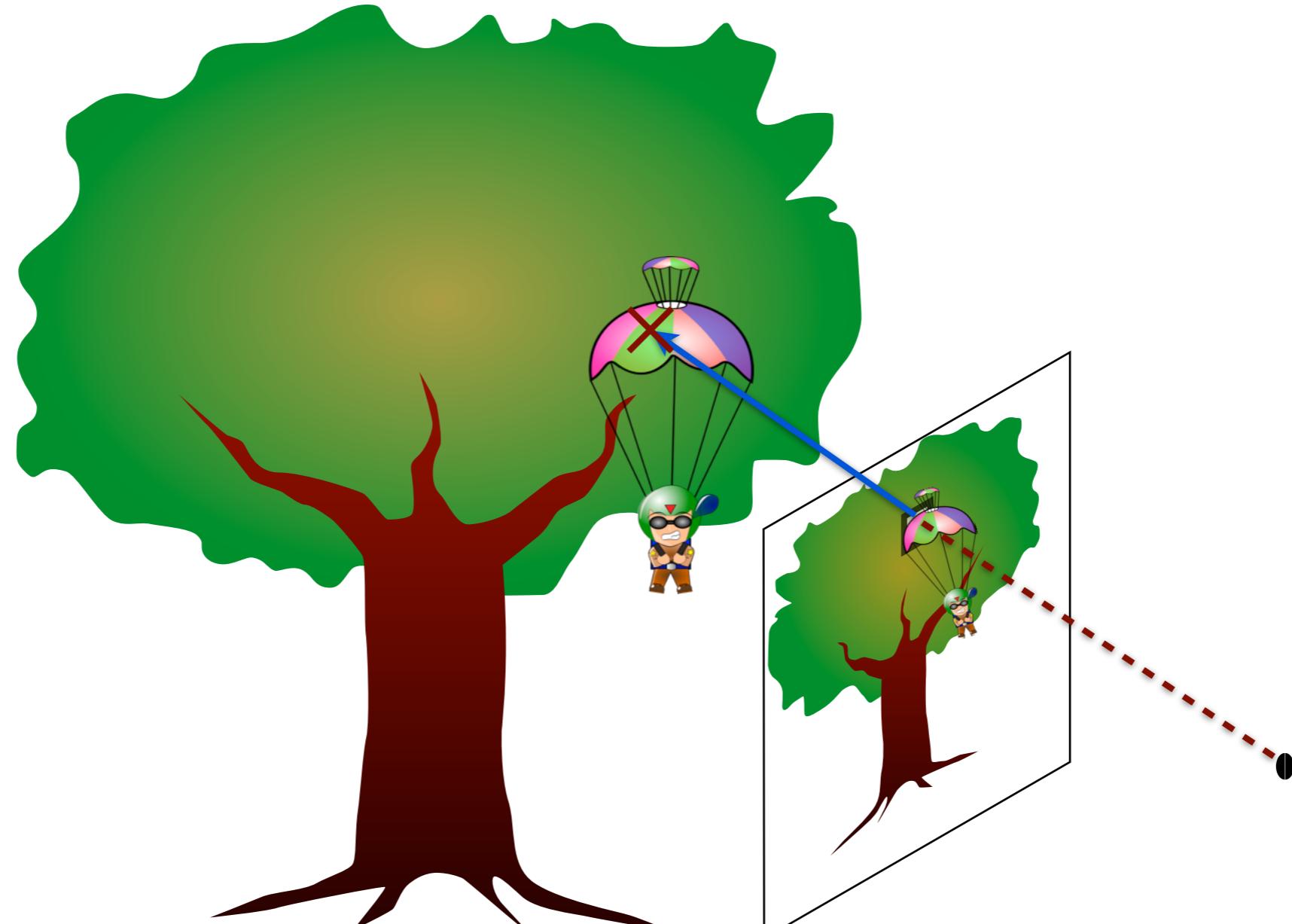
2D → 3D

# Raytracing



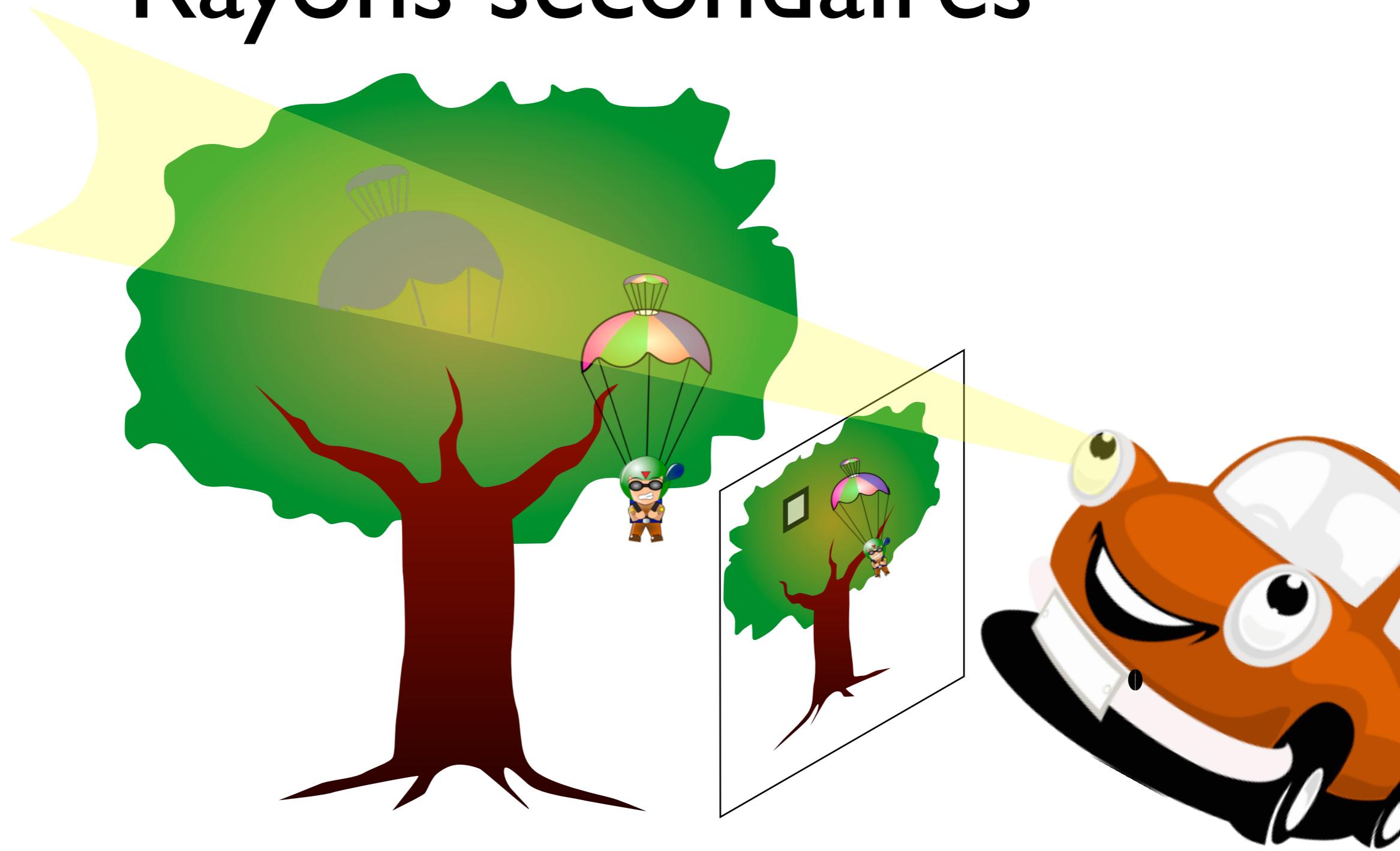
2D → 3D

# Raytracing



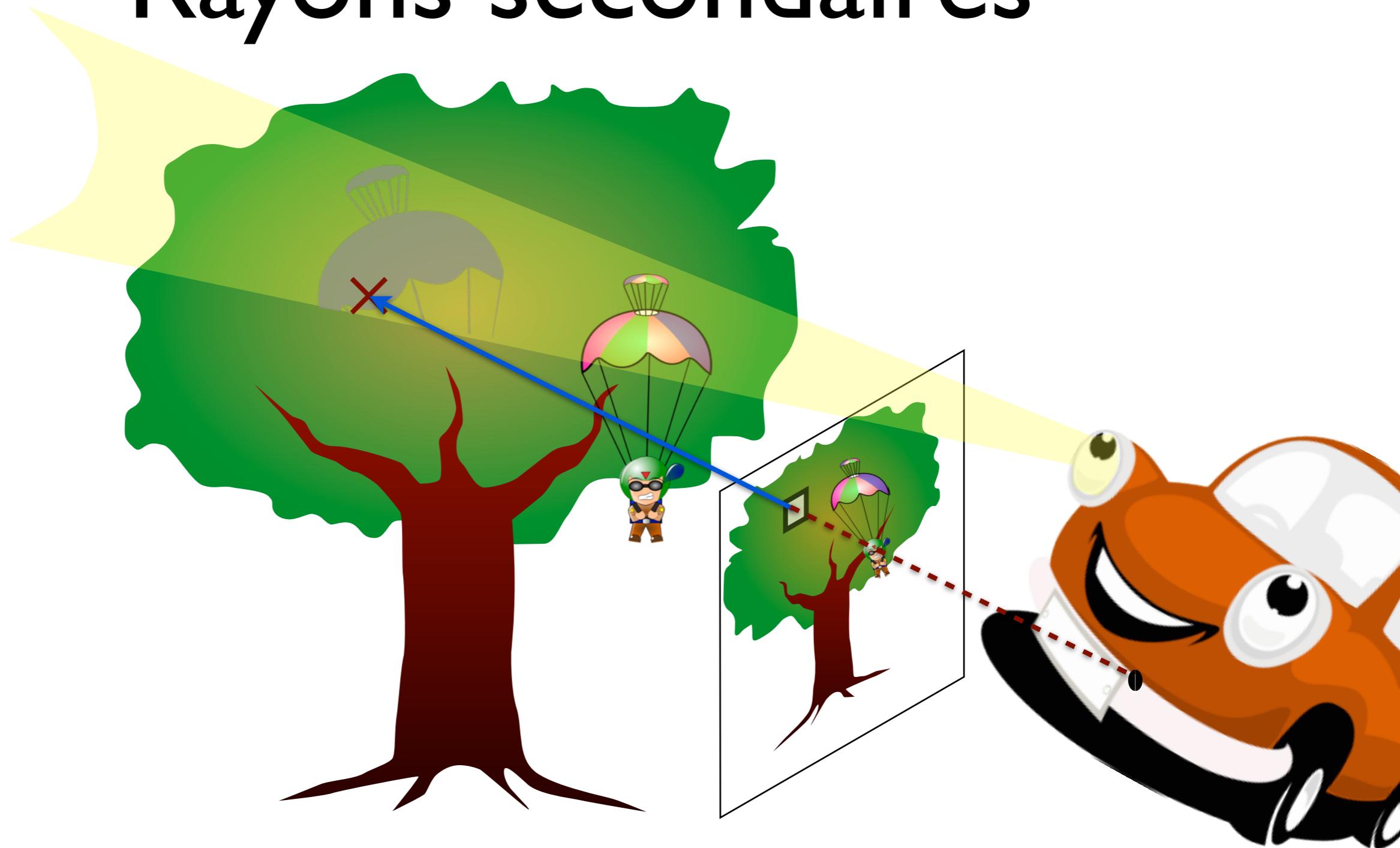
2D → 3D

# Rayons secondaires



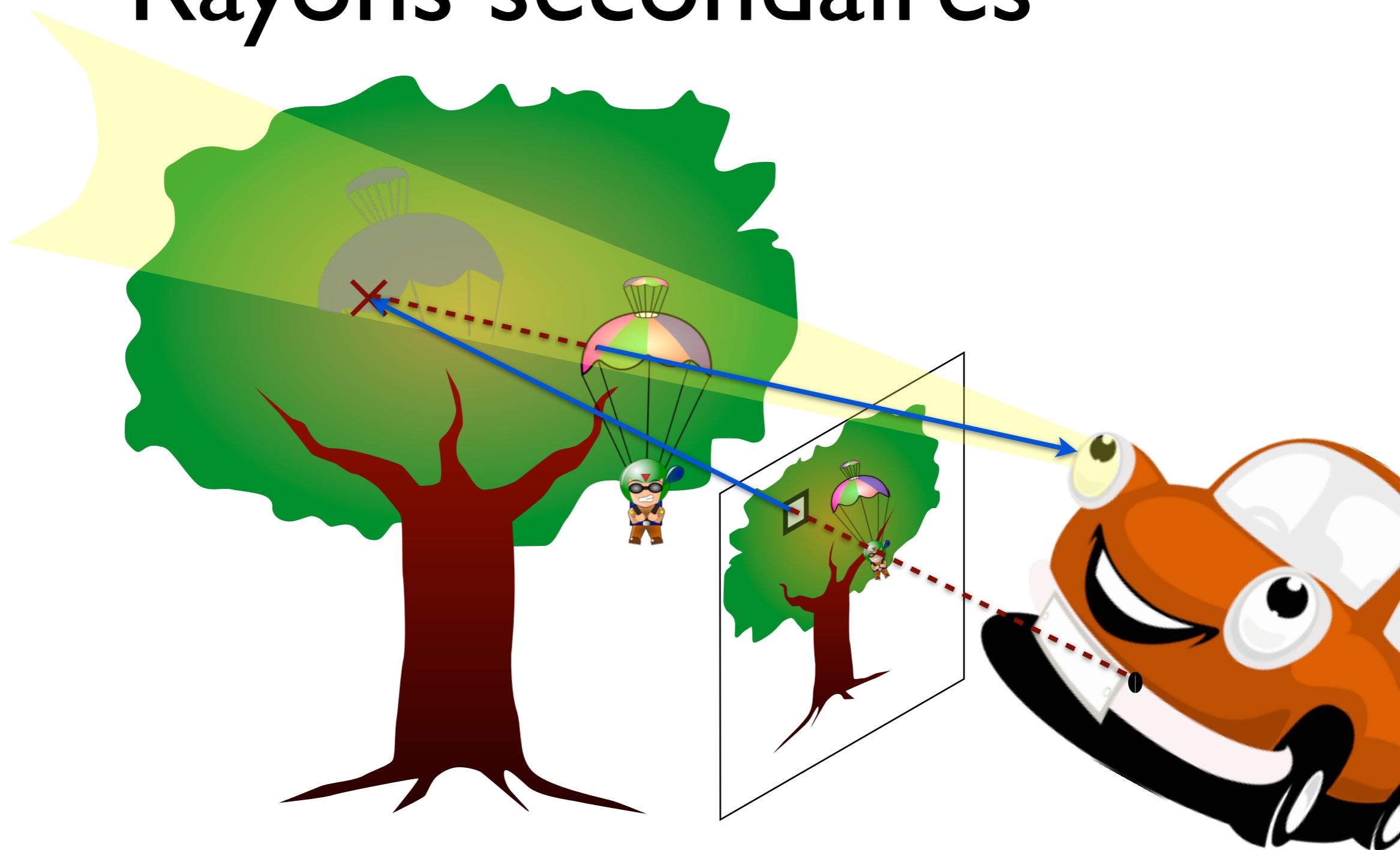
cast shadows, reflexions, refractions

# Rayons secondaires



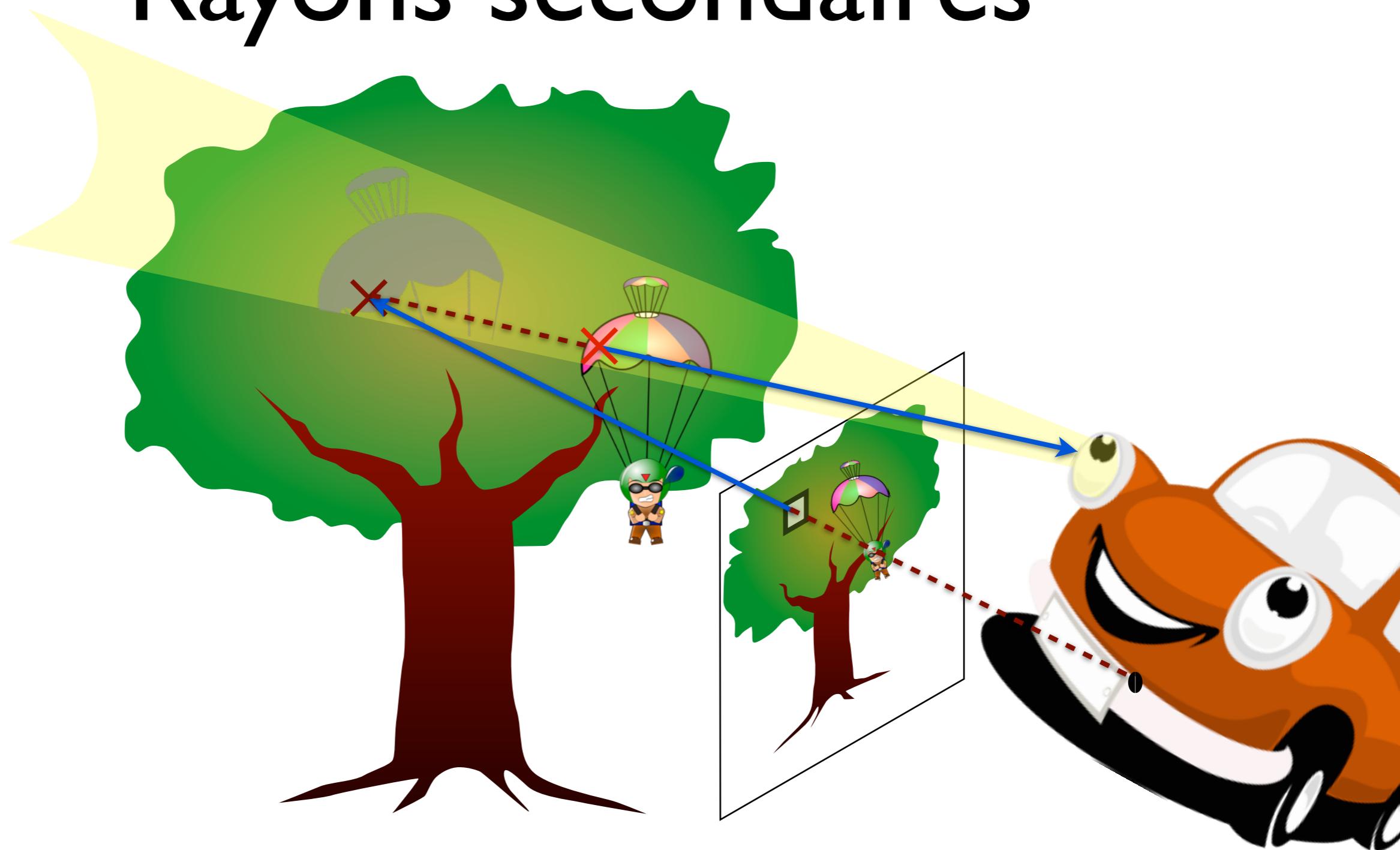
cast shadows, reflexions, refractions

# Rayons secondaires



cast shadows, reflexions, refractions

# Rayons secondaires



cast shadows, reflexions, refractions

# Exemples



# Exemples



Ombres douces

# Exemples

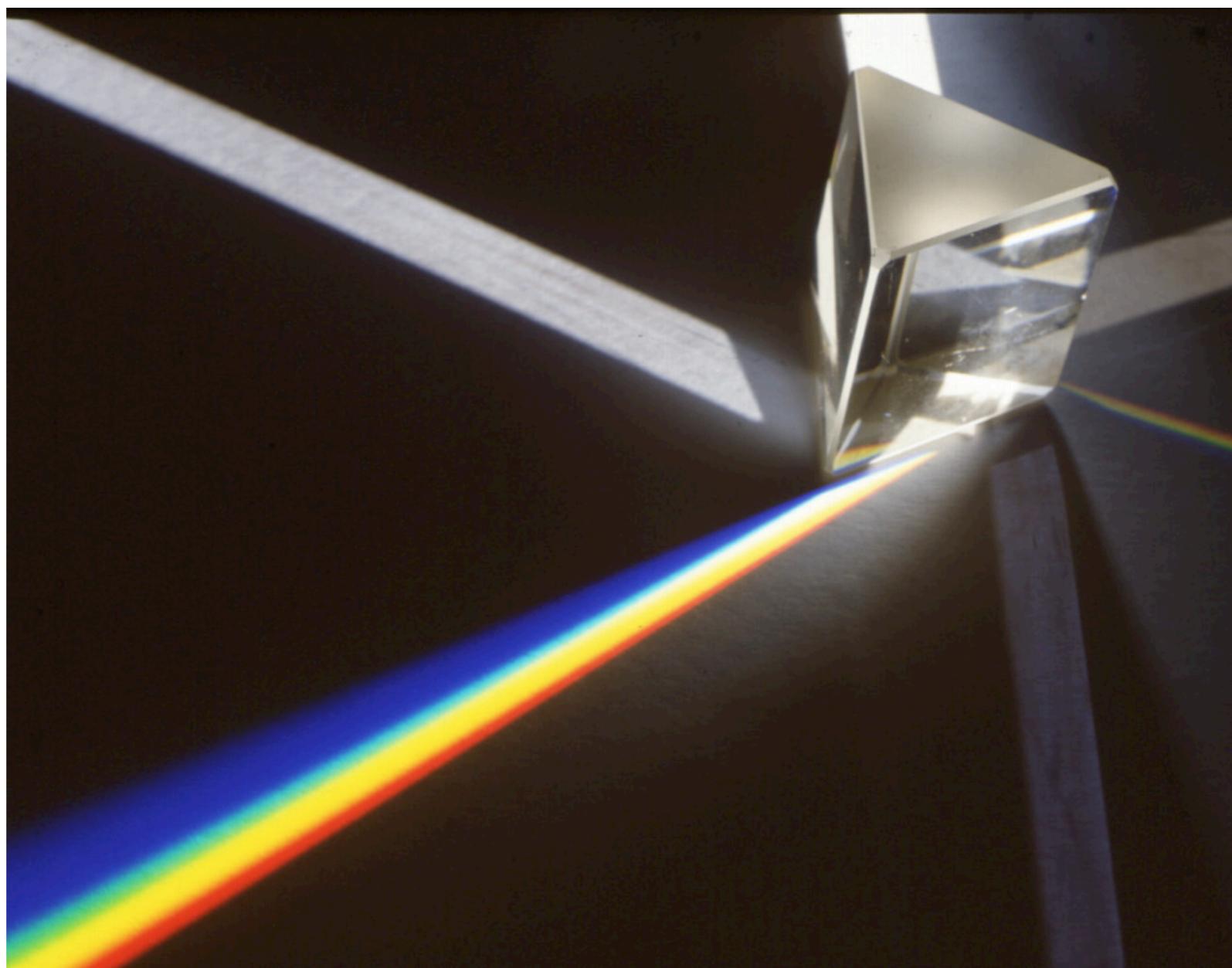


# Exemples

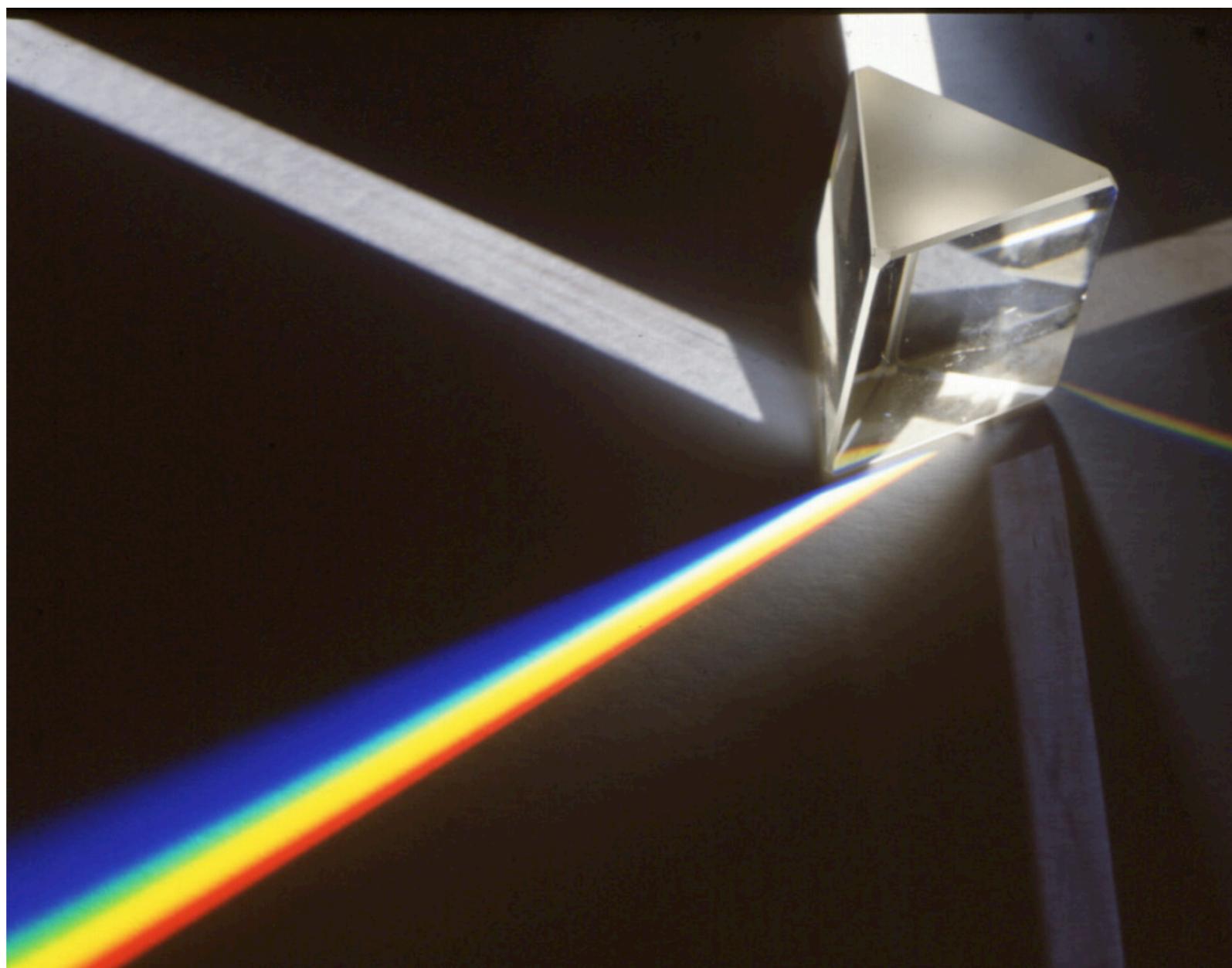


Profondeur de champs

# Exemples

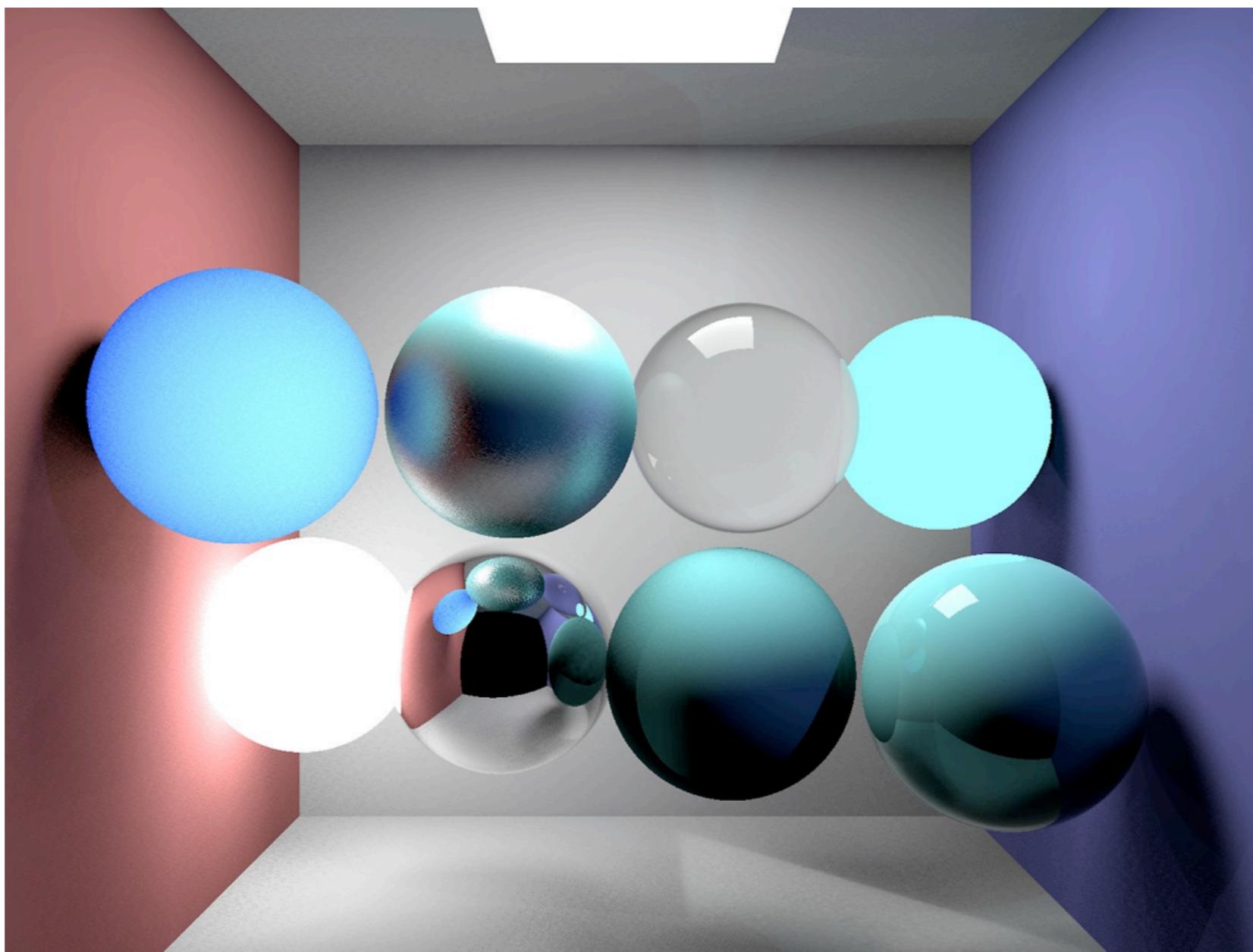


# Exemples



**RGB → multispectral**

# Exemples

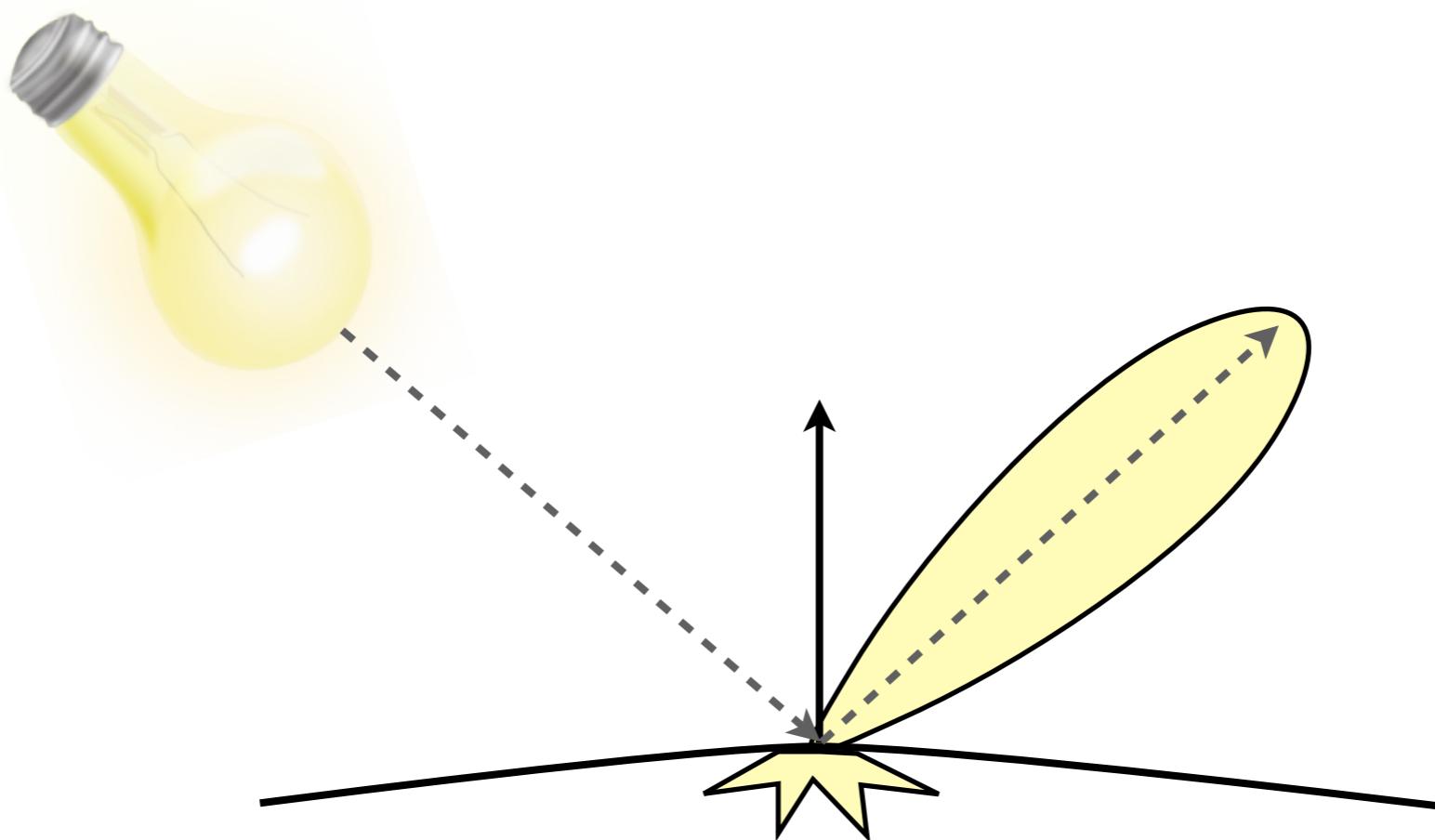


En quoi ces sphères diffèrent-elles ?

# Matériaux

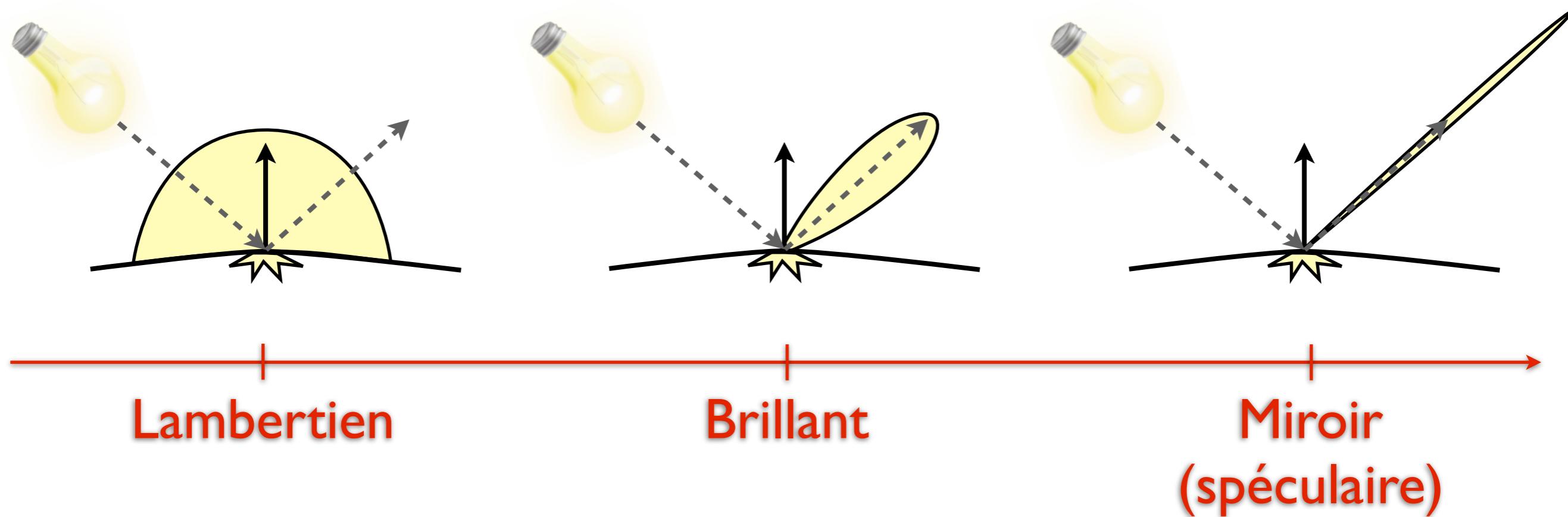
reflectance,...

# Reflectance

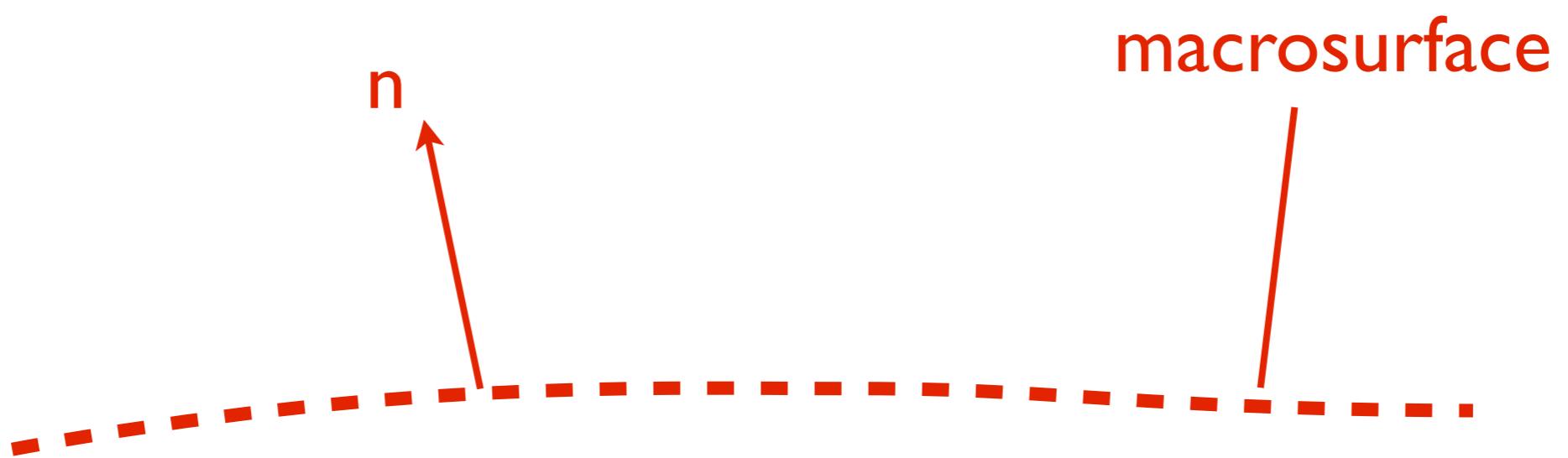


BRDF

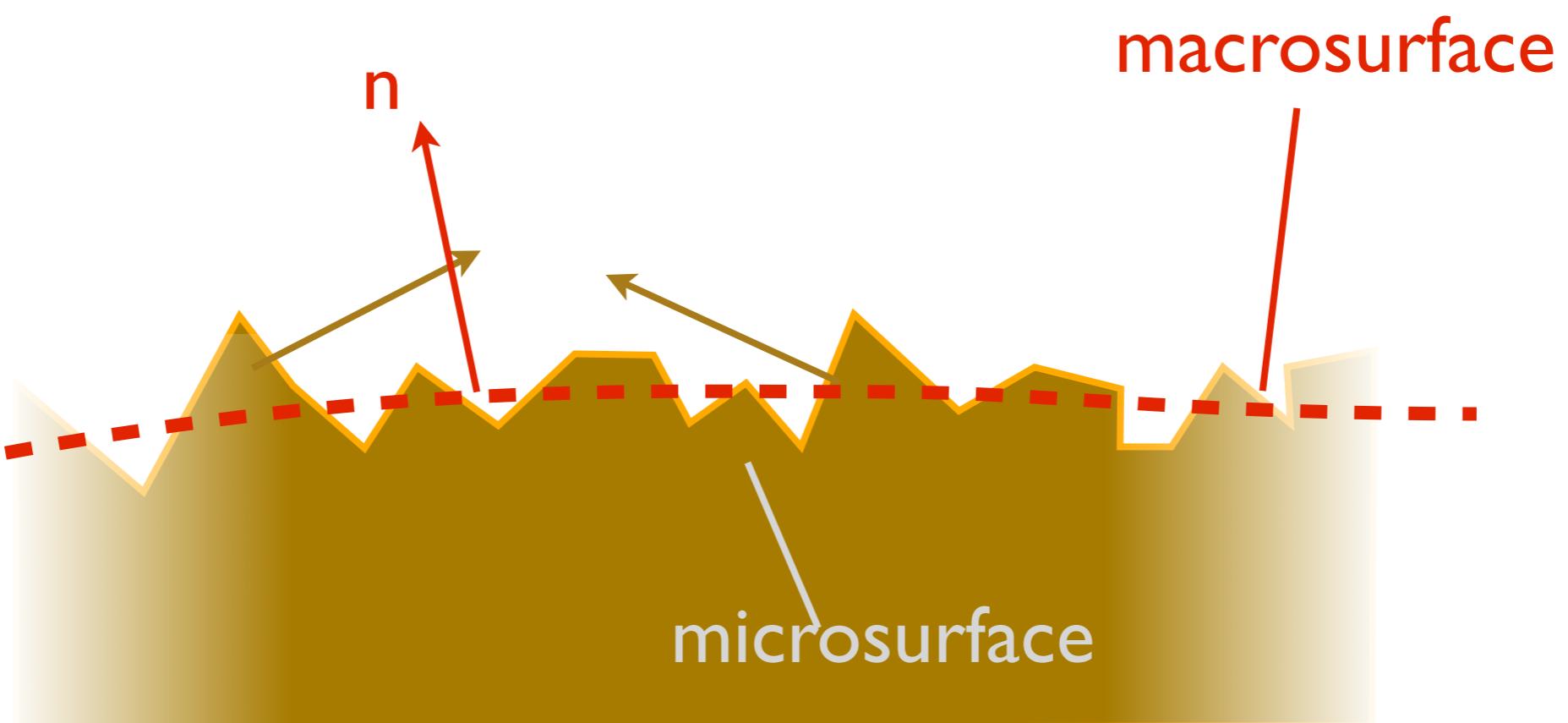
# Reflectance



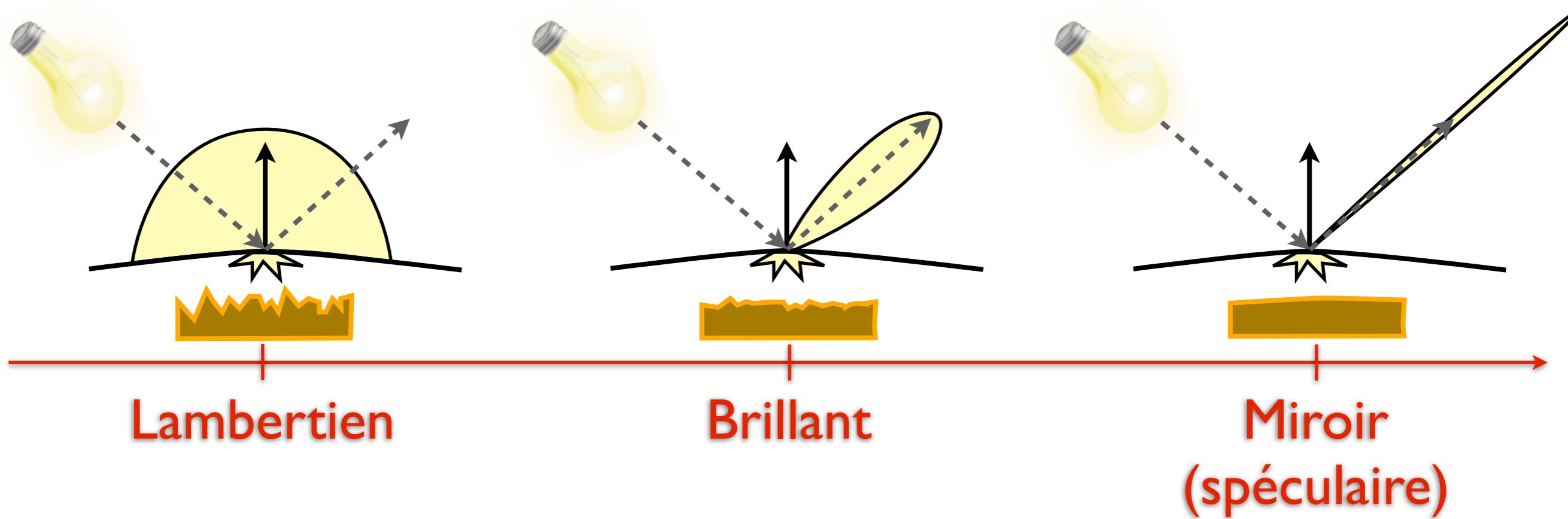
# Rugosité



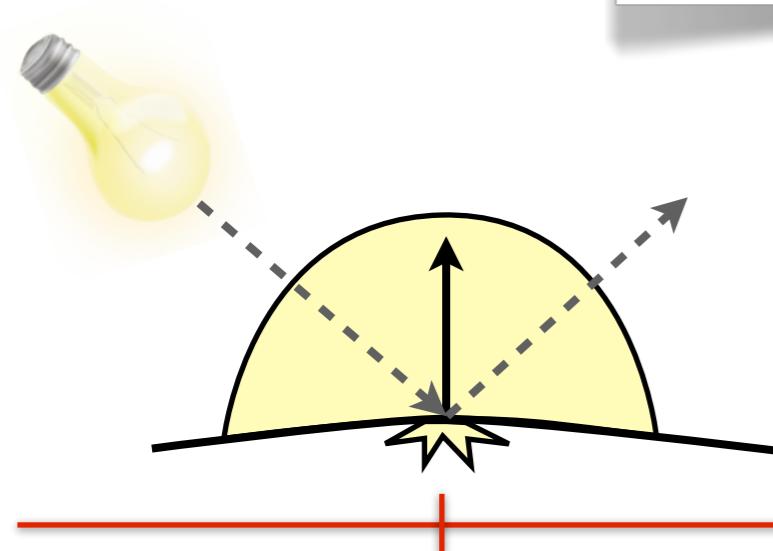
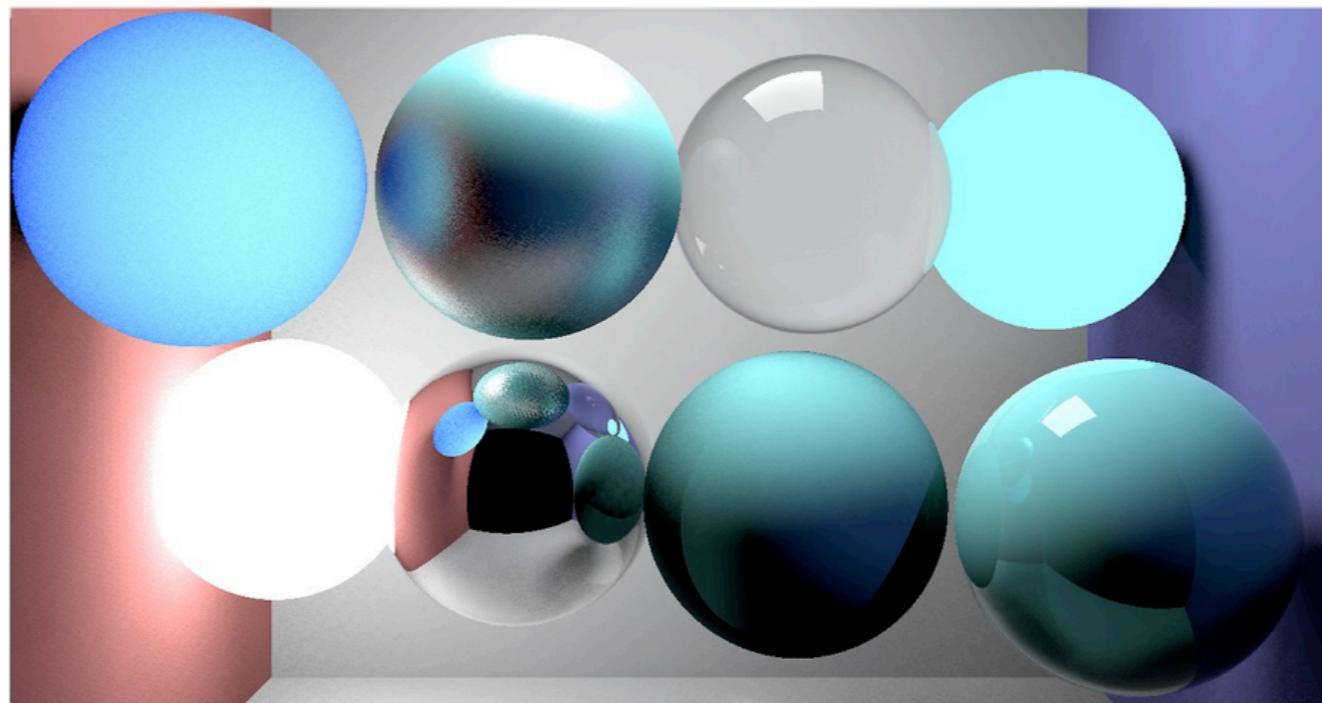
# Rugosité



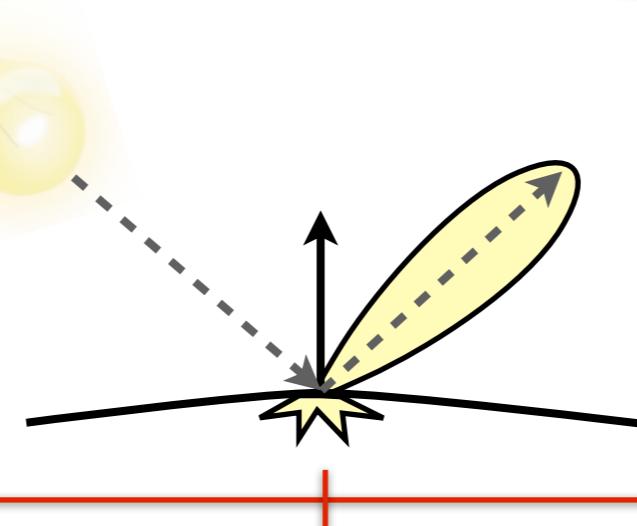
# Rugosité



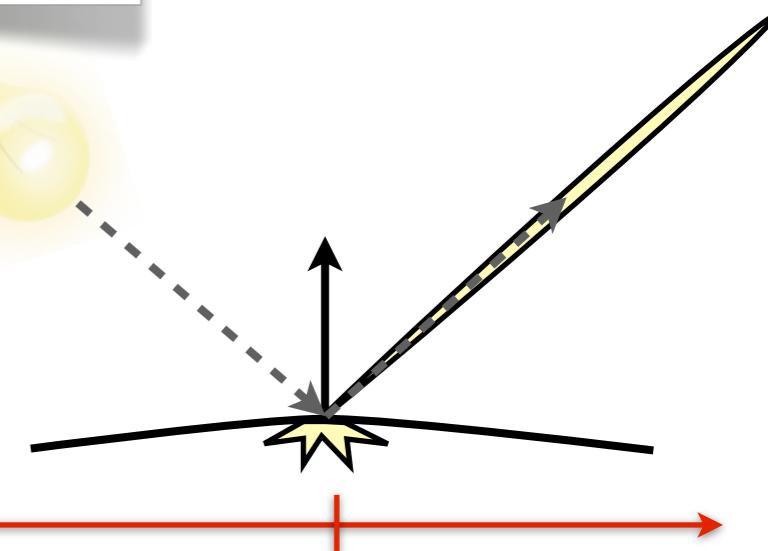
# Qui est qui?



Lambertien

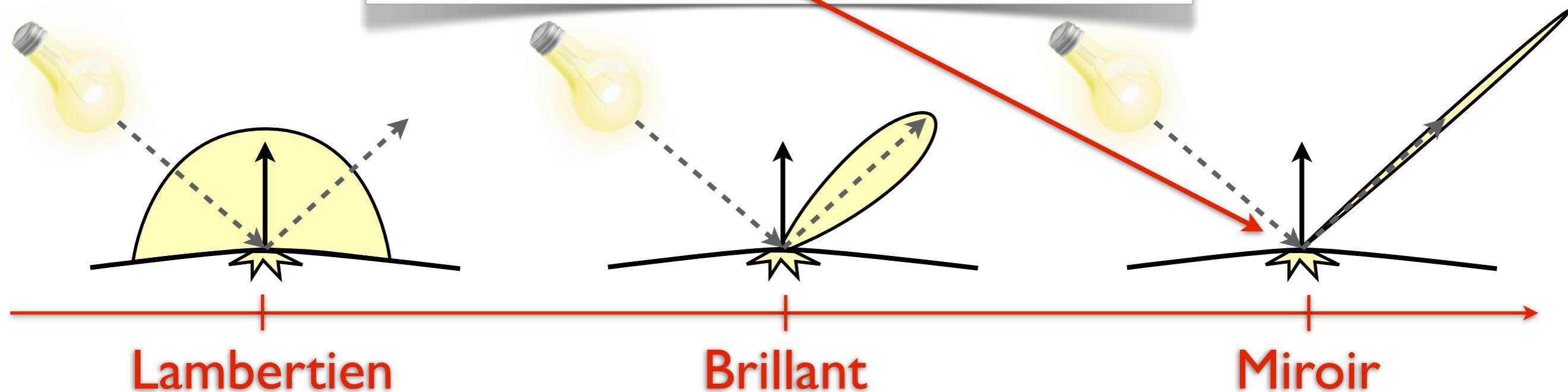
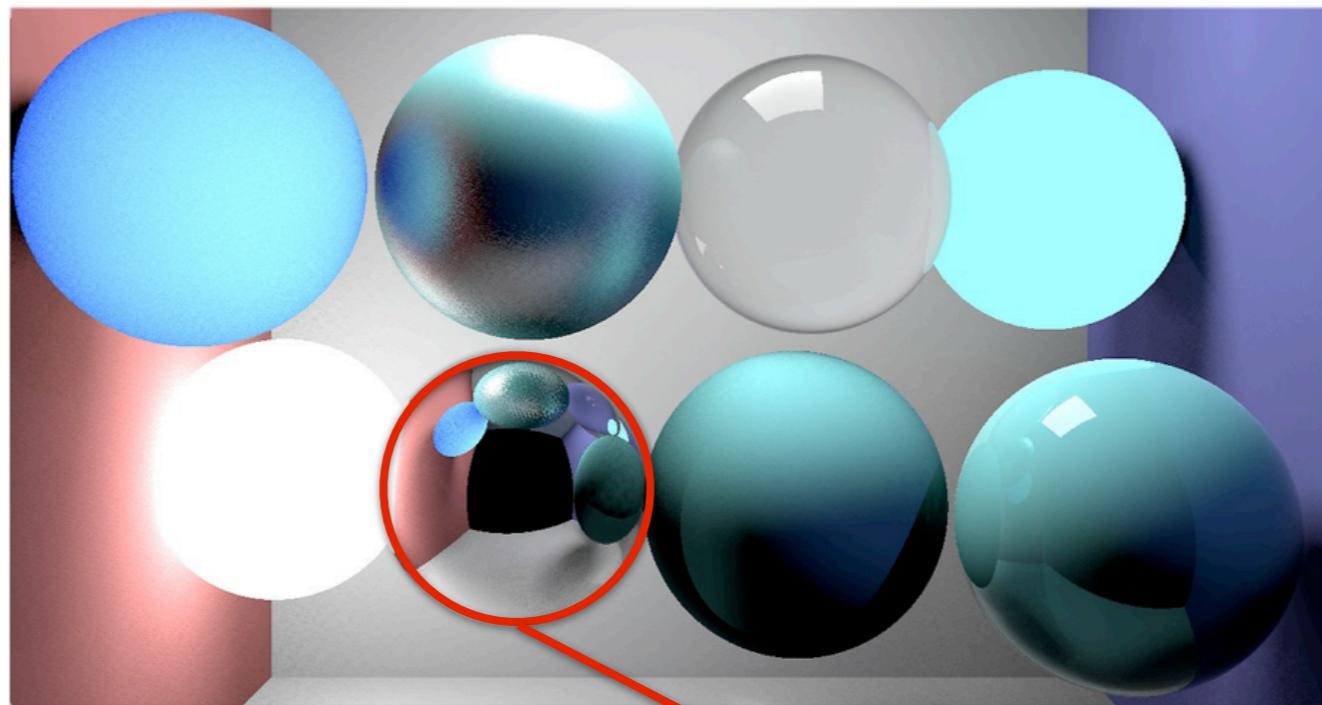


Brillant



Miroir

# Qui est qui?

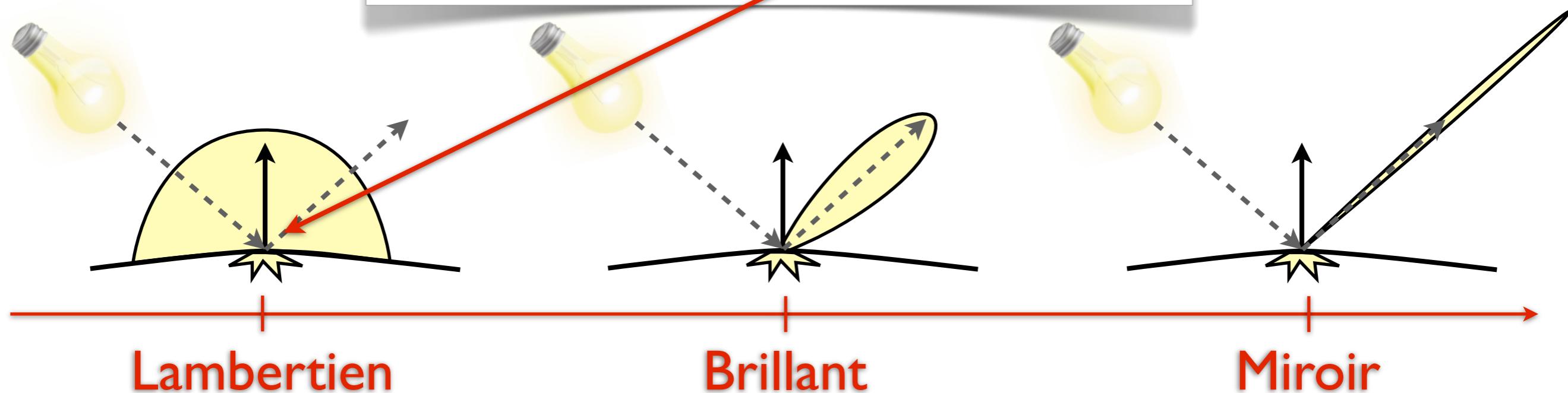
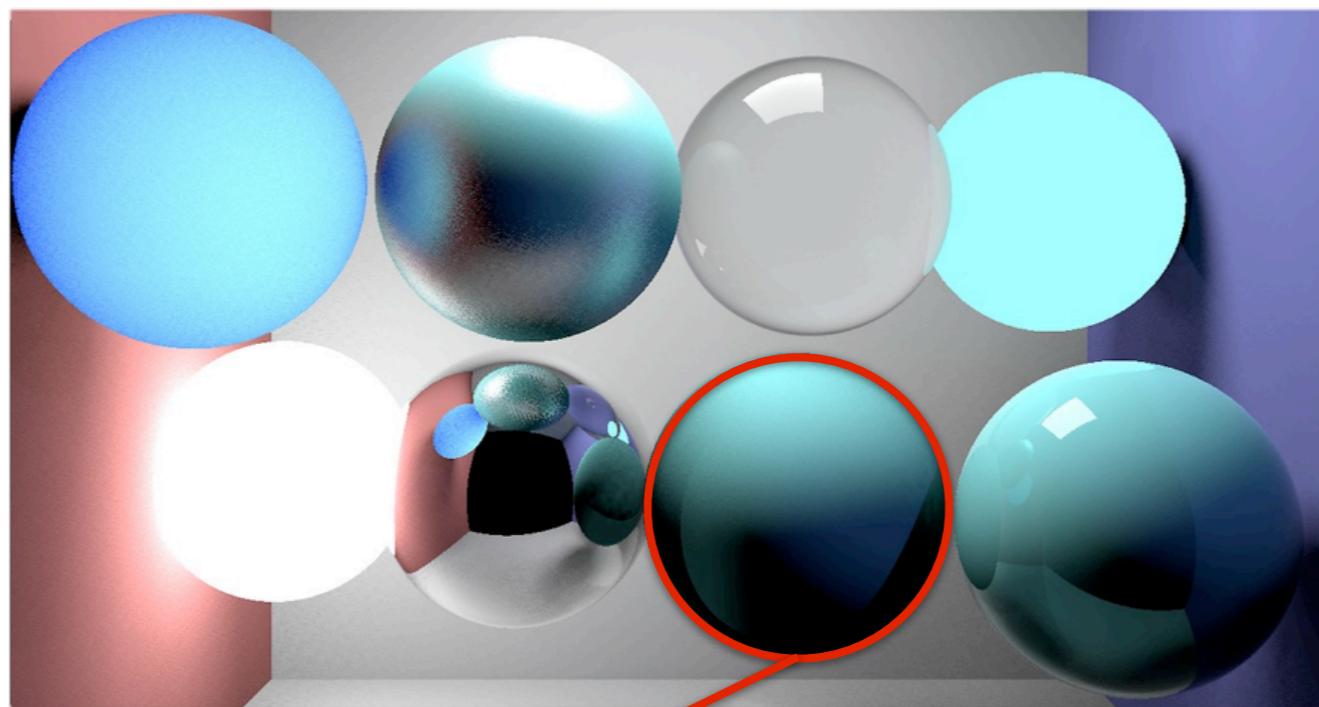


Lambertien

Brillant

Miroir

# Qui est qui?

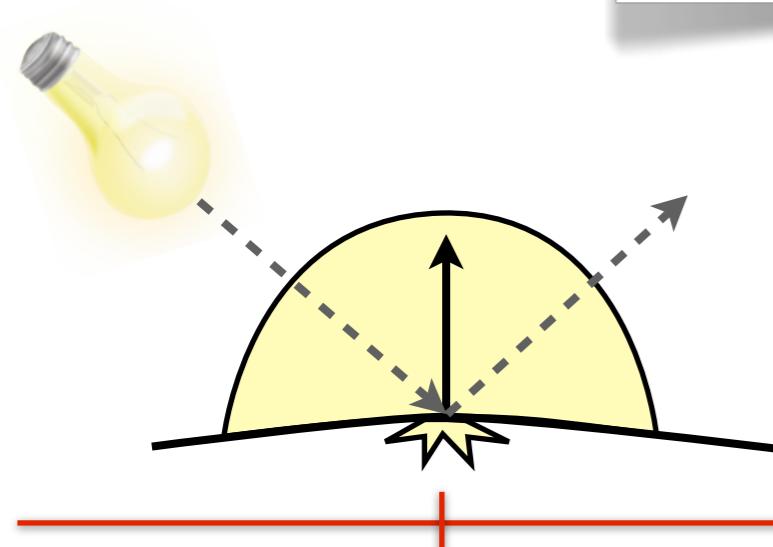
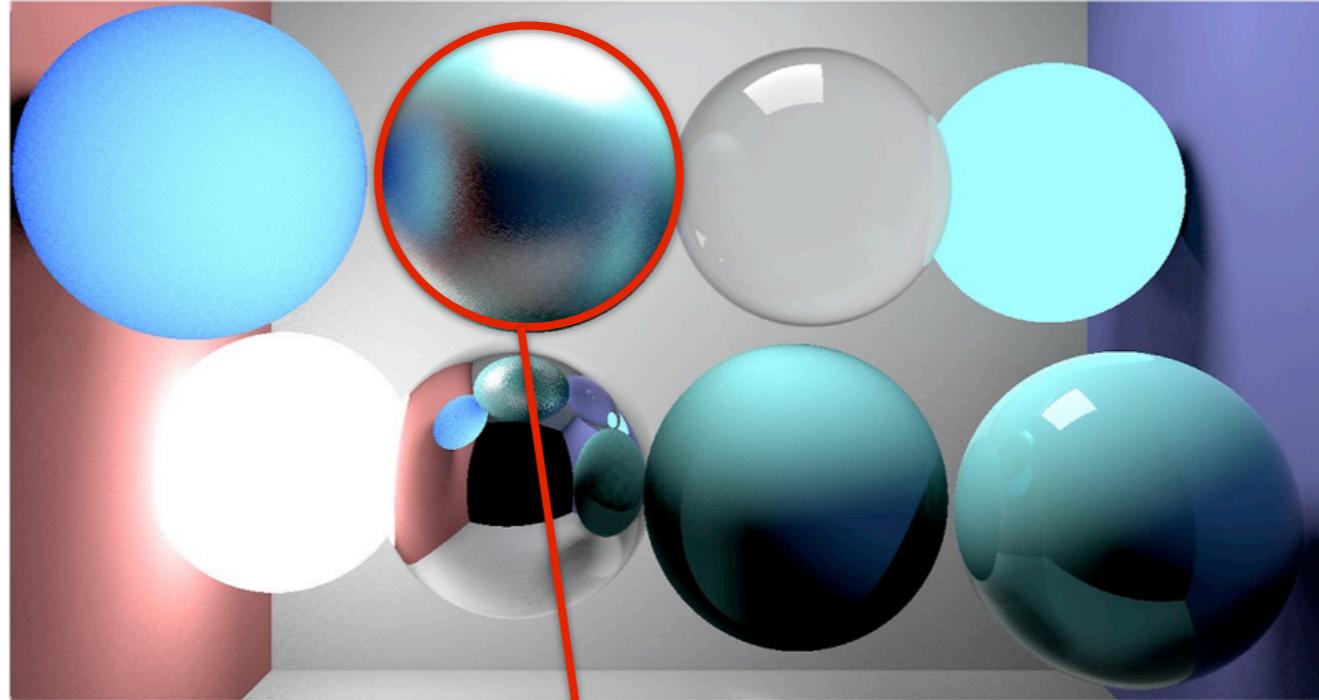


Lambertien

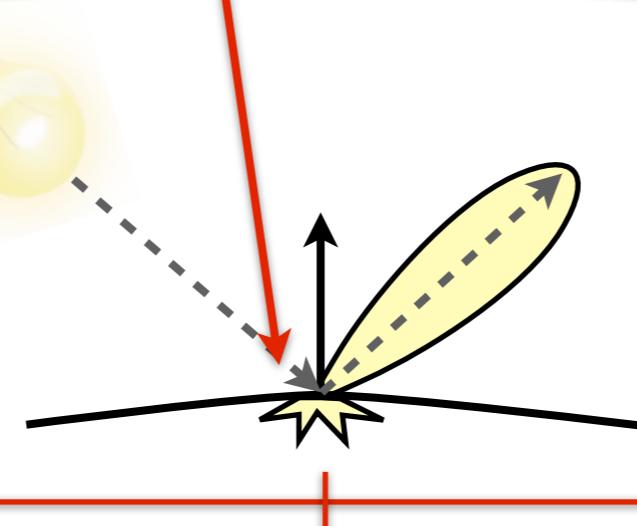
Brillant

Miroir

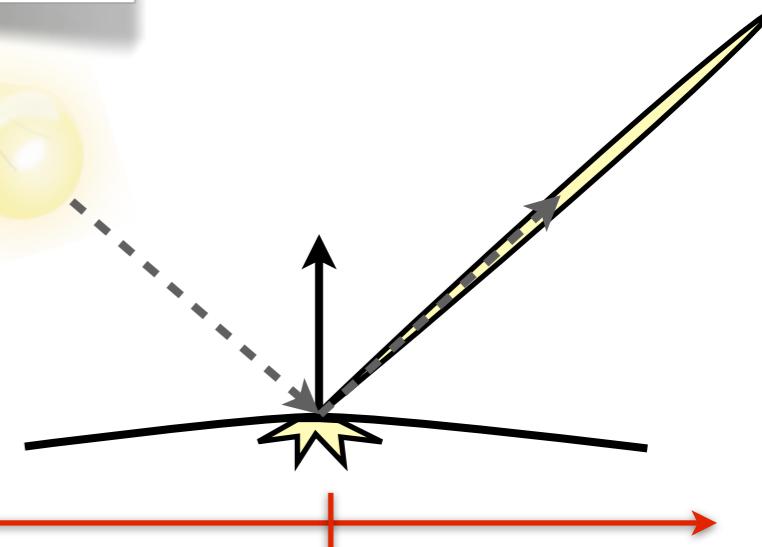
# Qui est qui?



Lambertien

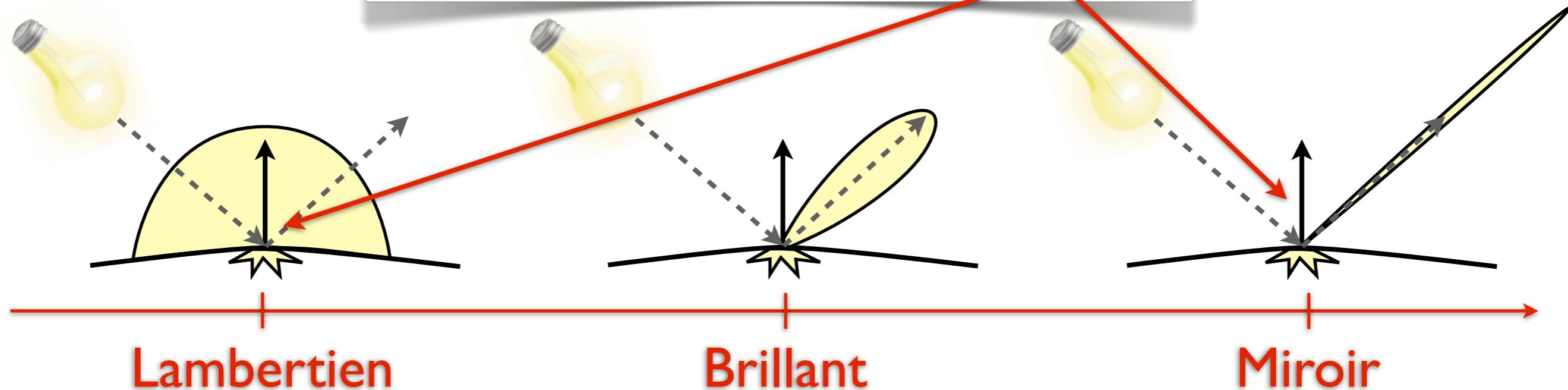
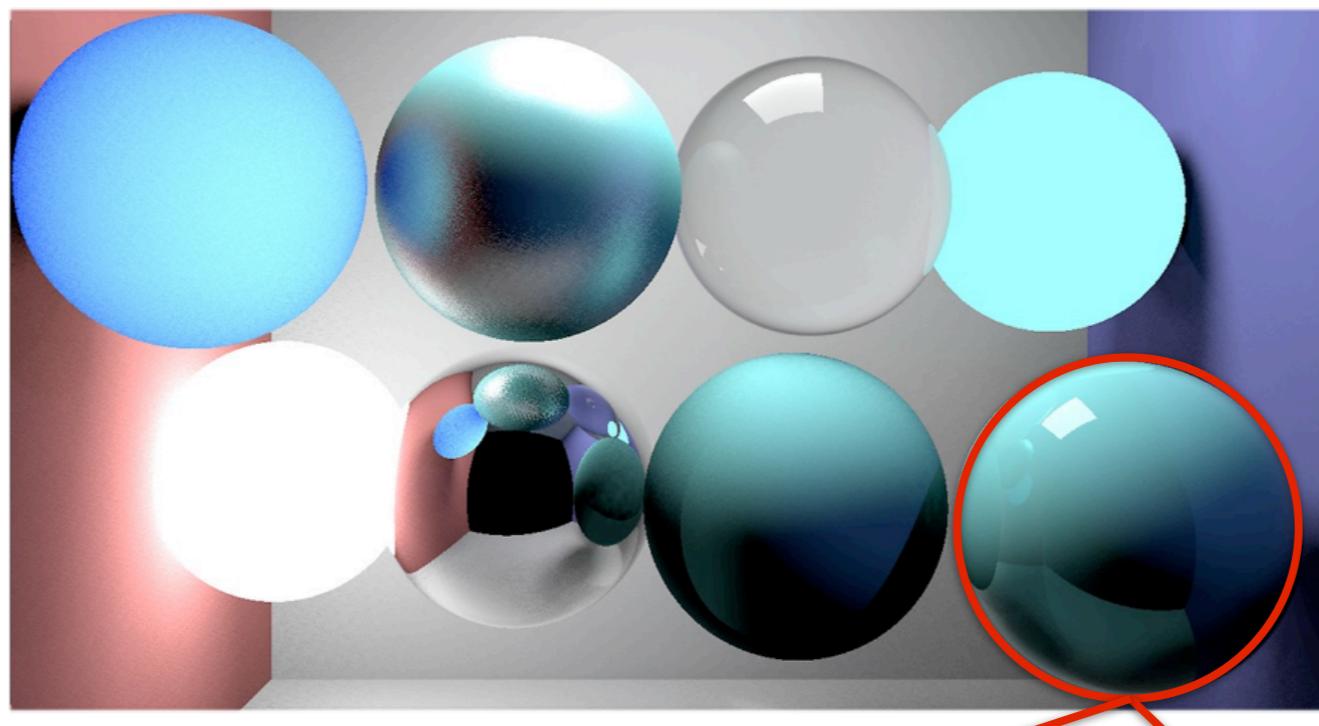


Brillant



Miroir

# Qui est qui?

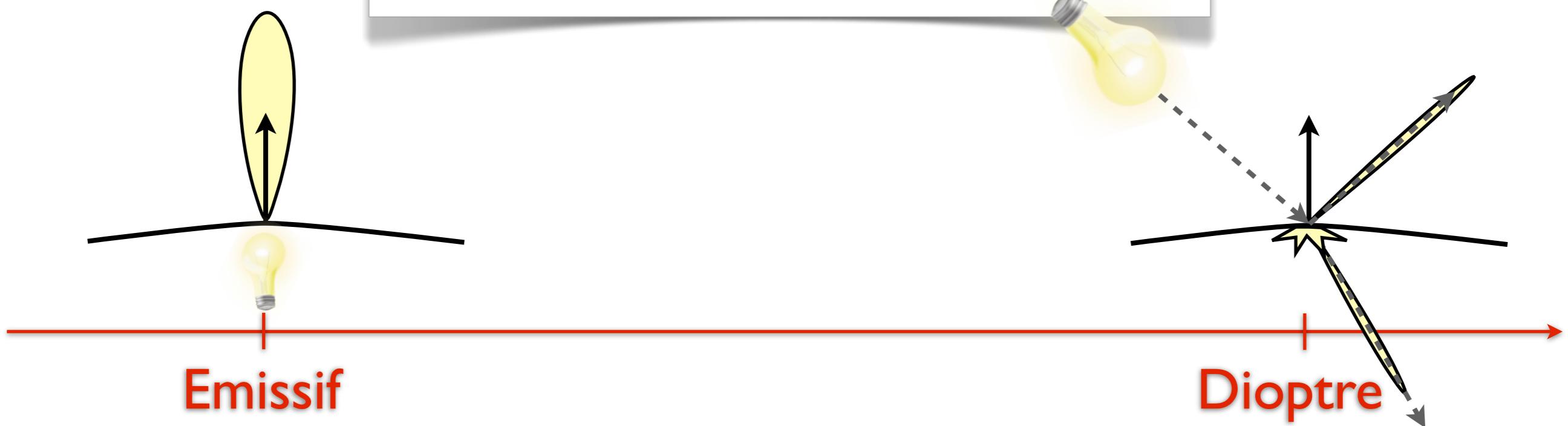
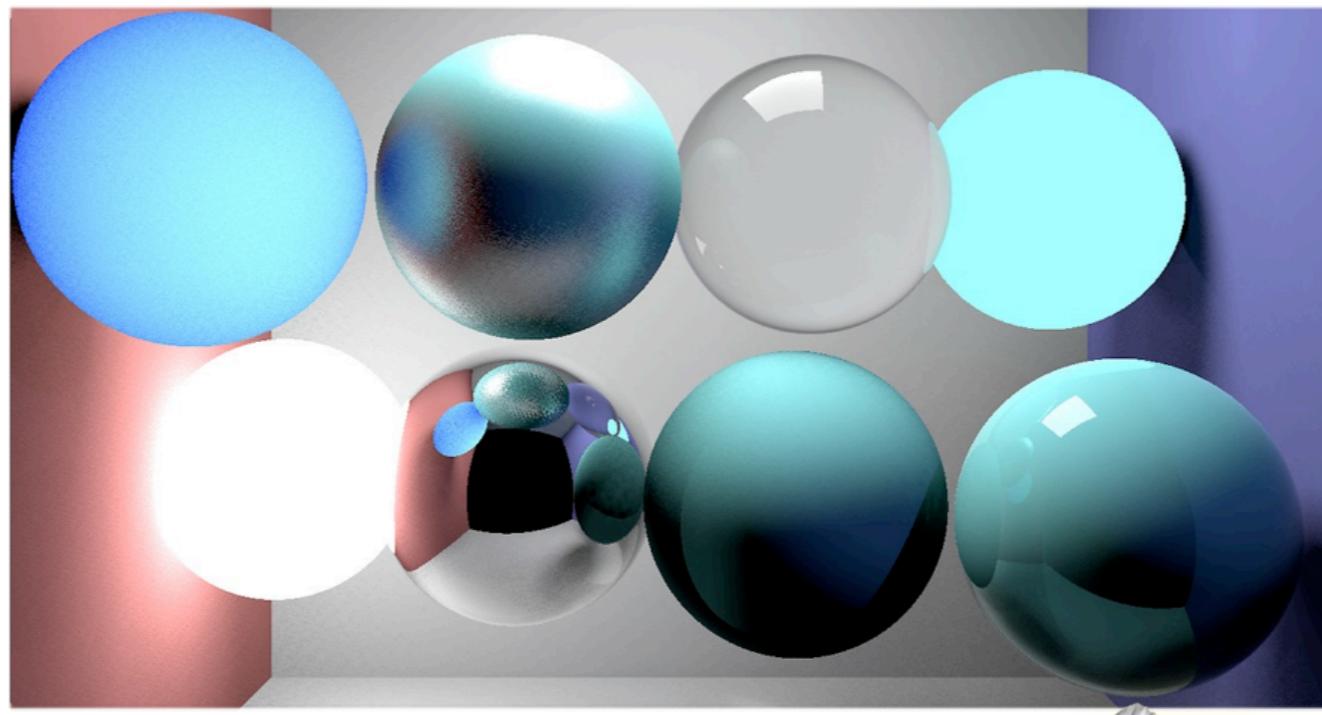


Lambertien

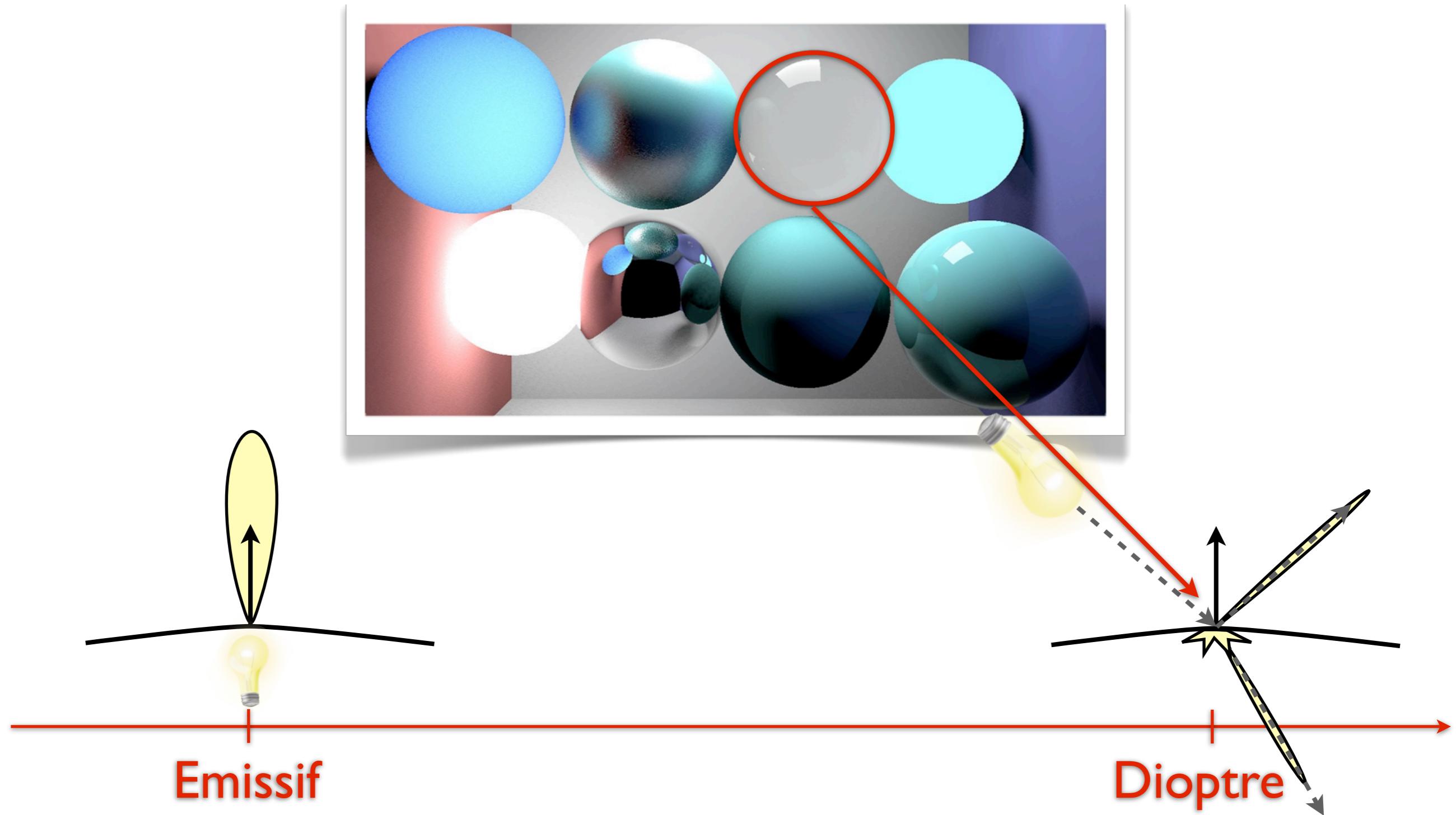
Brillant

Miroir

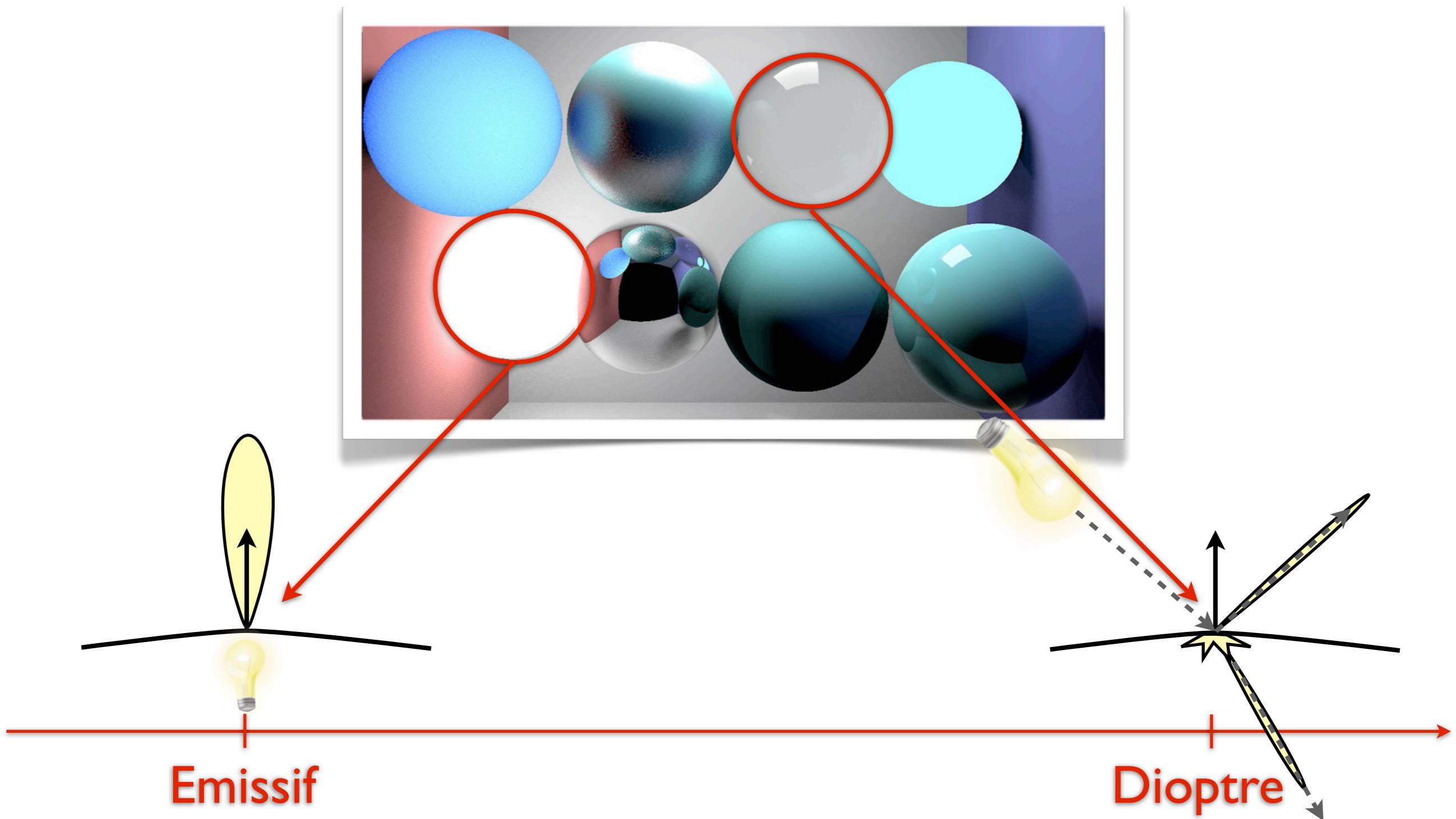
# Qui est qui?



# Qui est qui?



# Qui est qui?



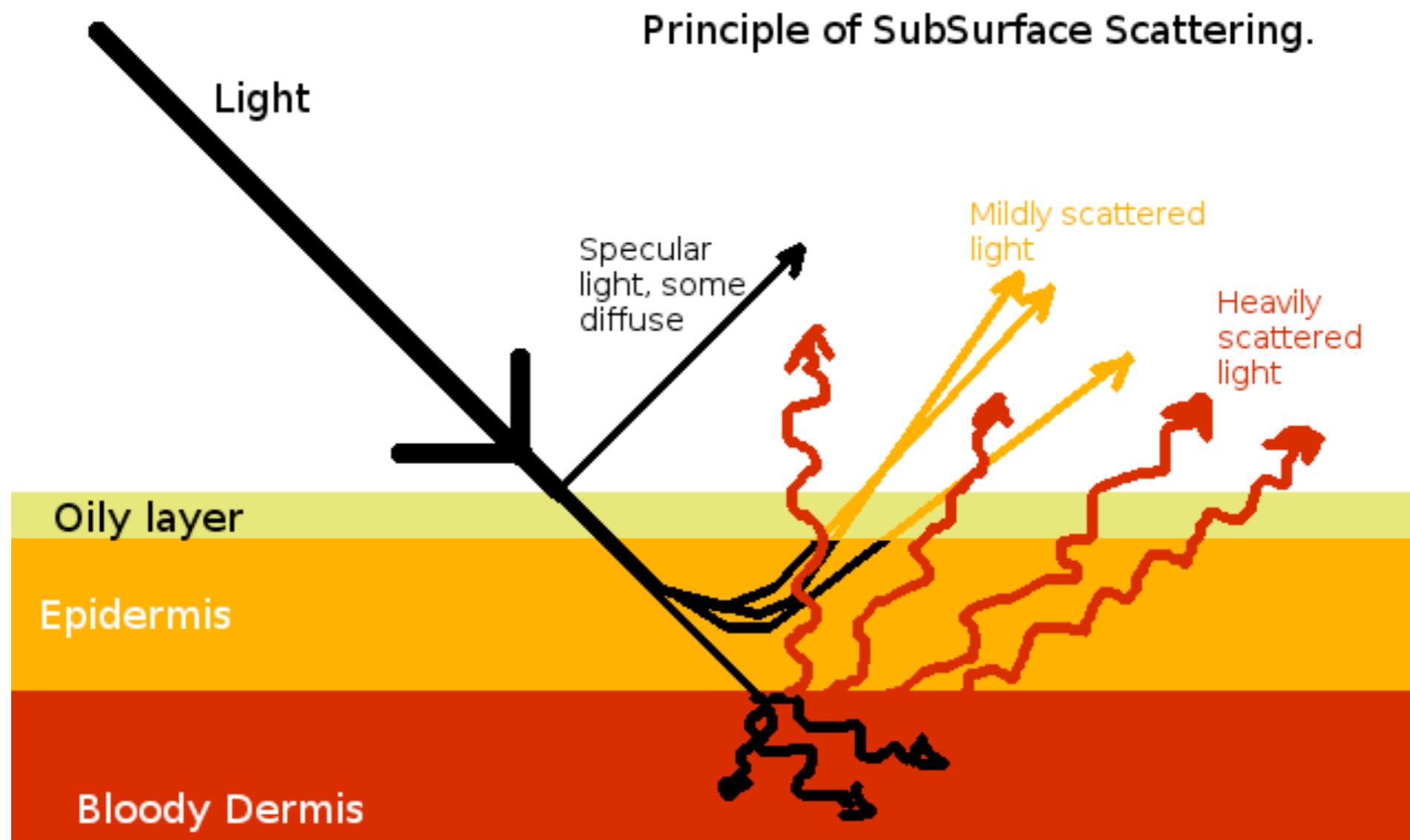
# Subsurface scattering

## Transluminescence



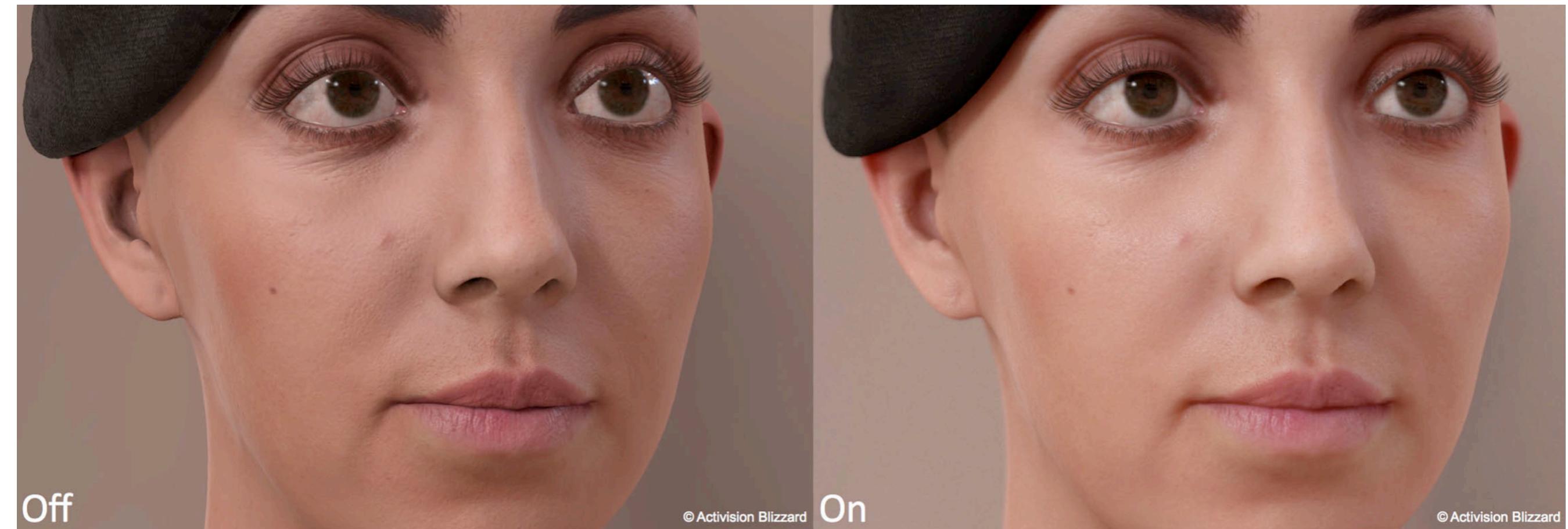
# Subsurface scattering

## Transluminescence



# Subsurface scattering

## Transluminescence



# Conclusion

# Notions abordées

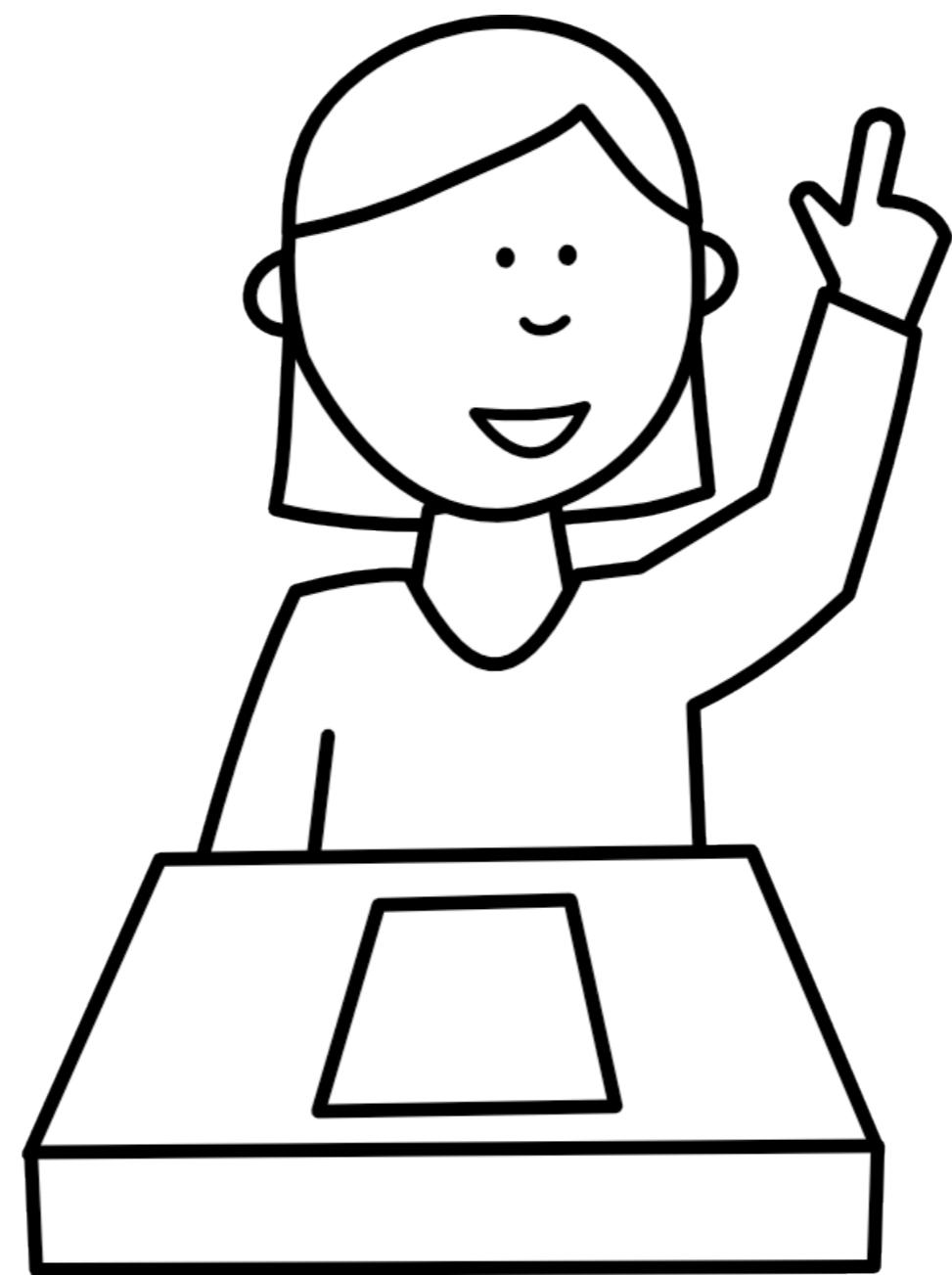
- Représentation
  - ✓ images
  - ✓ objets (maillage+texture+matériaux)
- Algorithmes
  - ✓ Rasterization (**fast** - **hacky**)
  - ✓ Ray tracing (**accurate** - **time consuming**)

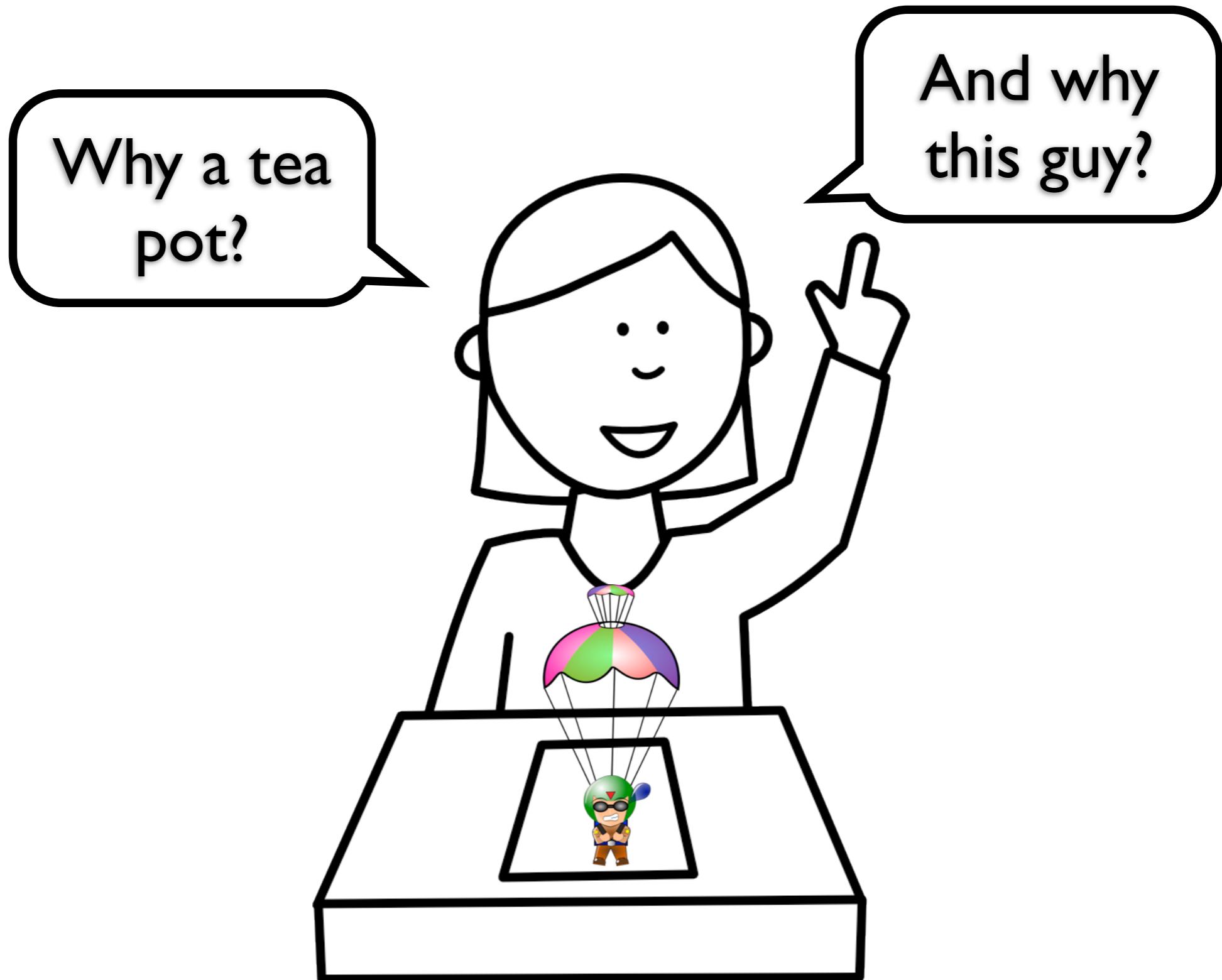
# Pour aller plus loin



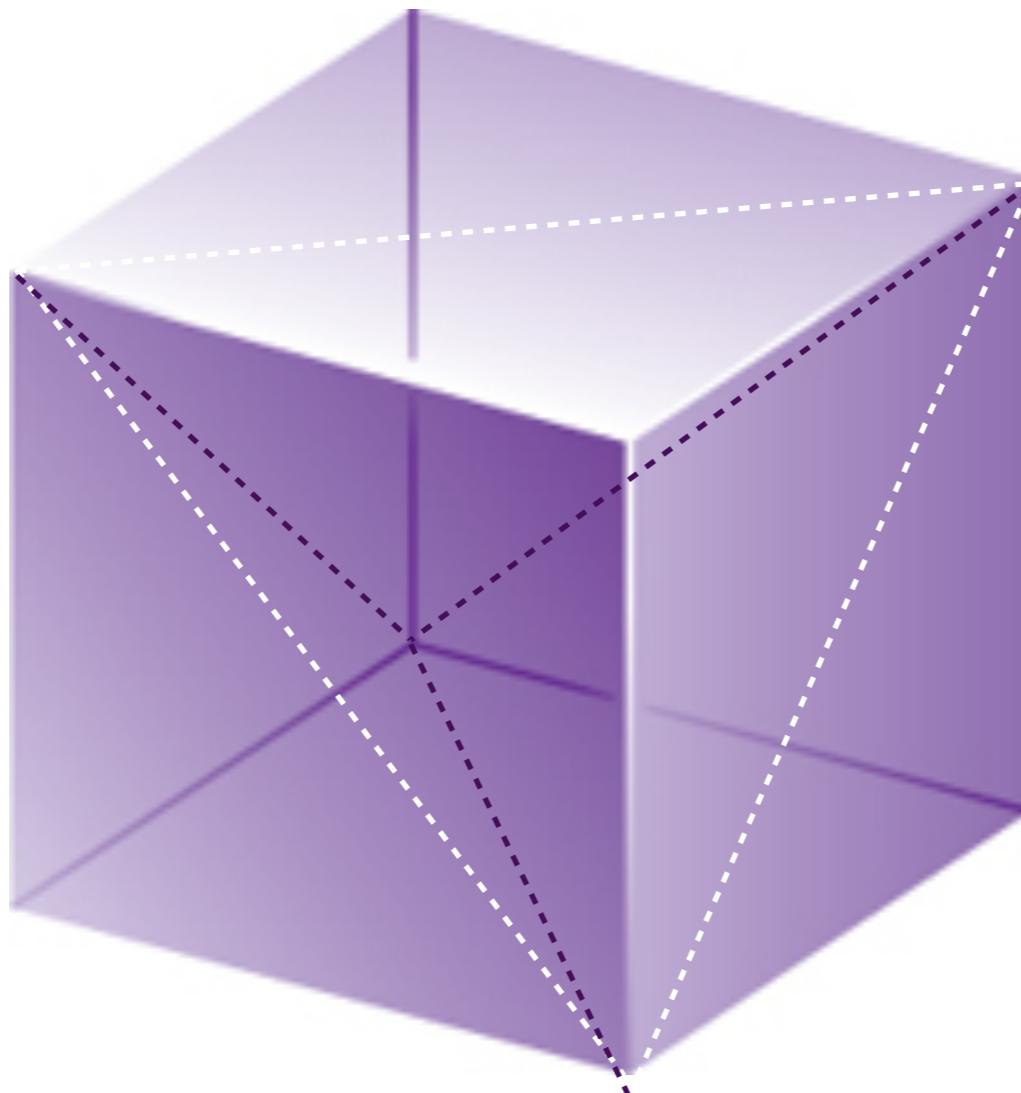
# Bibliographie

- **Pharr, M., and Humphreys, G.**  
*Physically based rendering: From theory to implementation.* Morgan Kaufmann, 2010.
- **Wright, Richard S., et al.**  
*OpenGL SuperBible: comprehensive tutorial and reference.* Pearson Education, 2010.

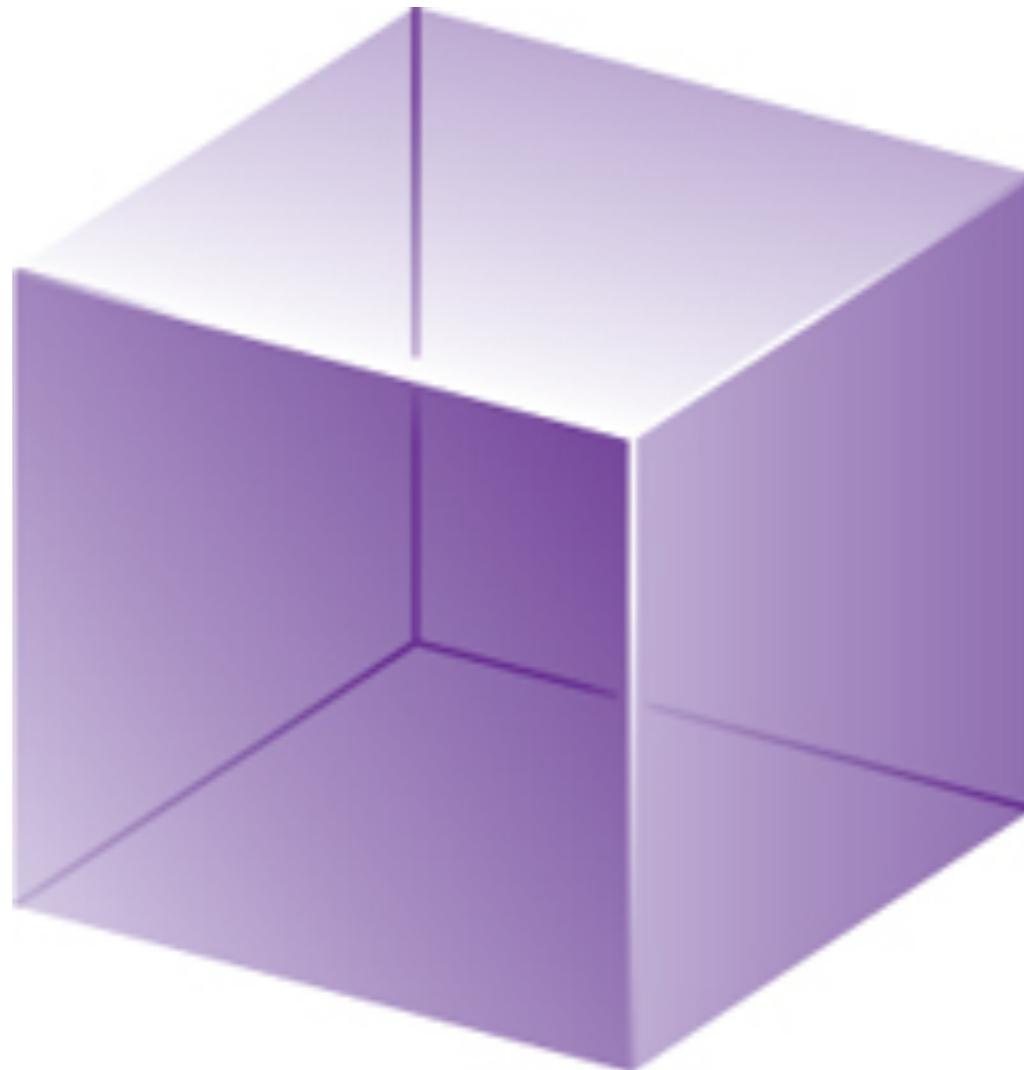




# Alternatives aux triangles

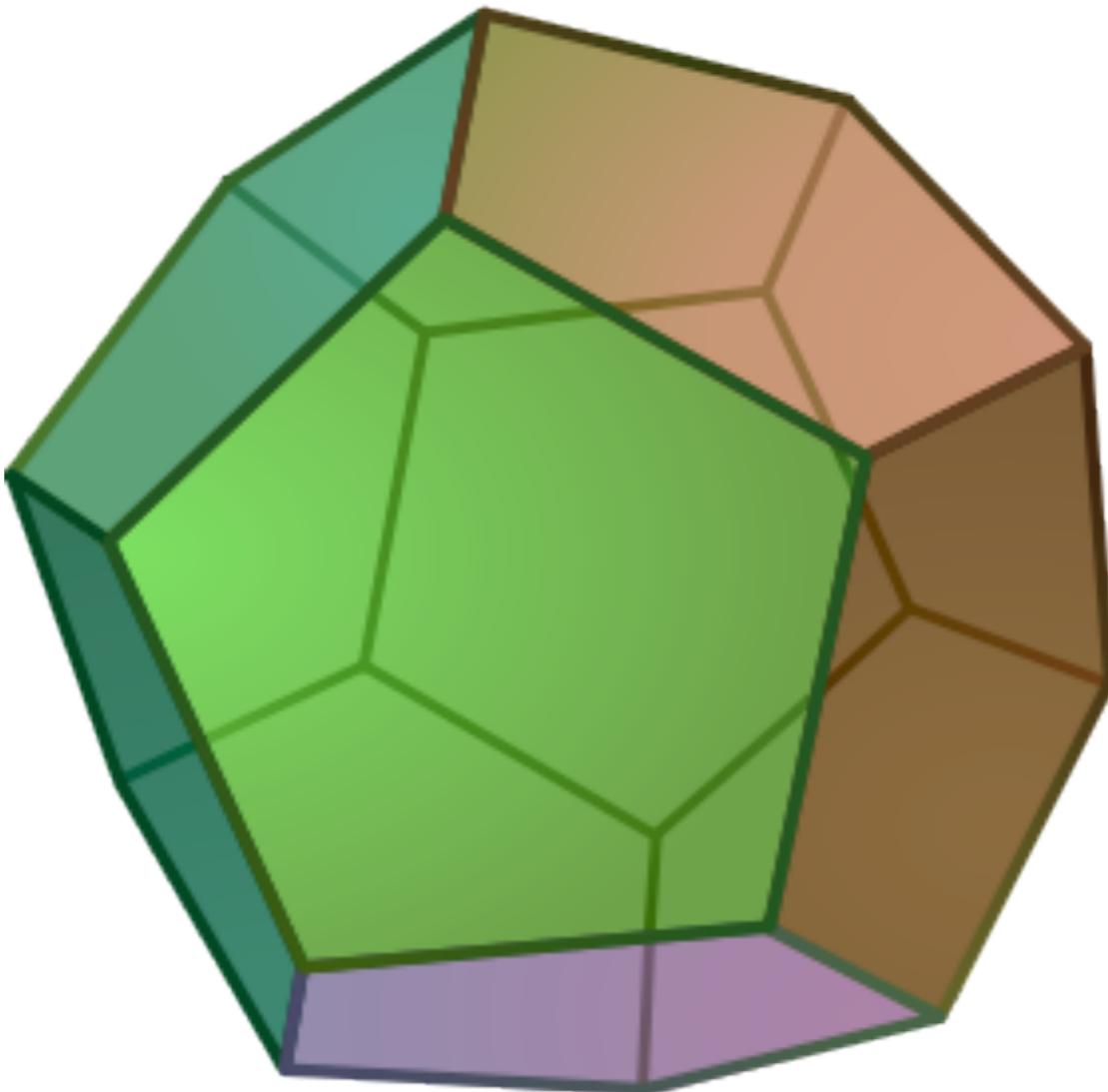


# Alternatives aux triangles



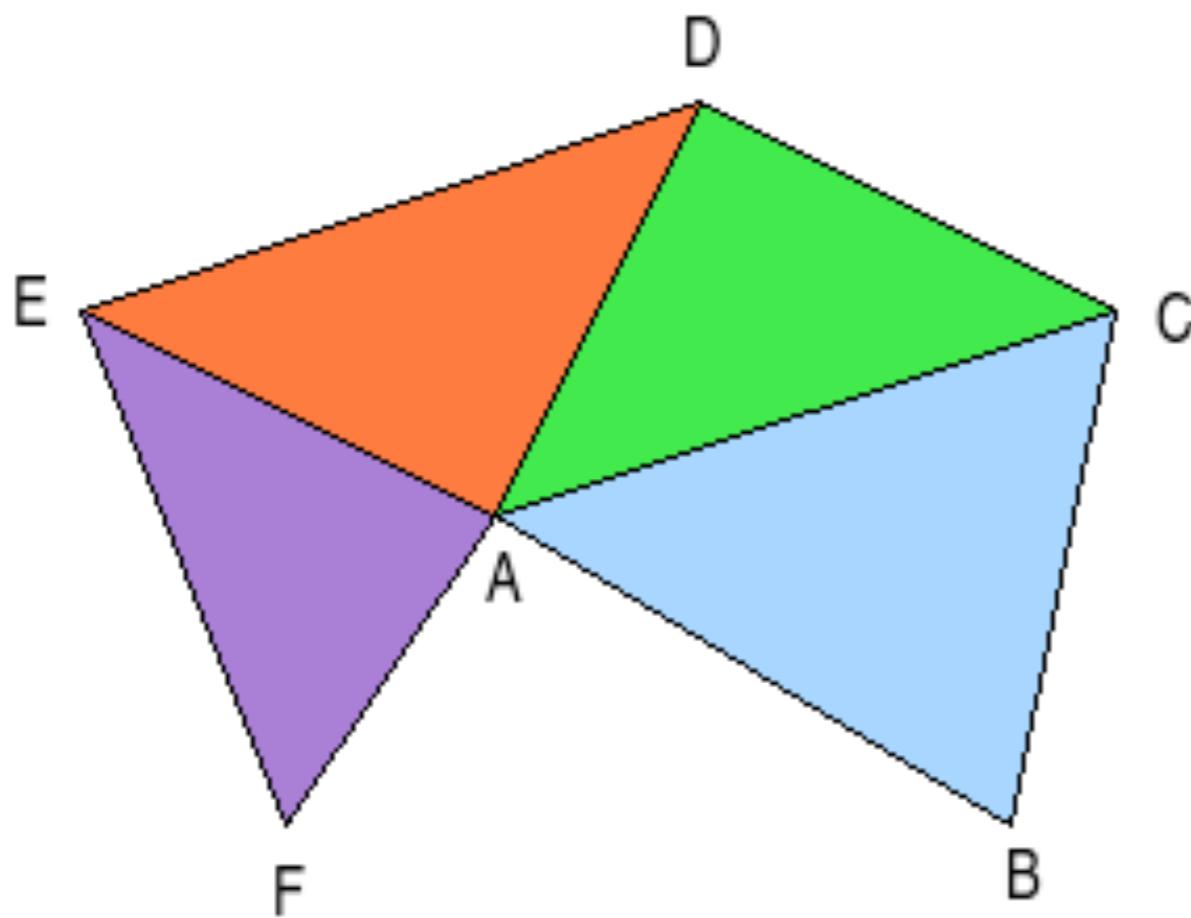
👉 **Quads**

# Alternatives aux triangles

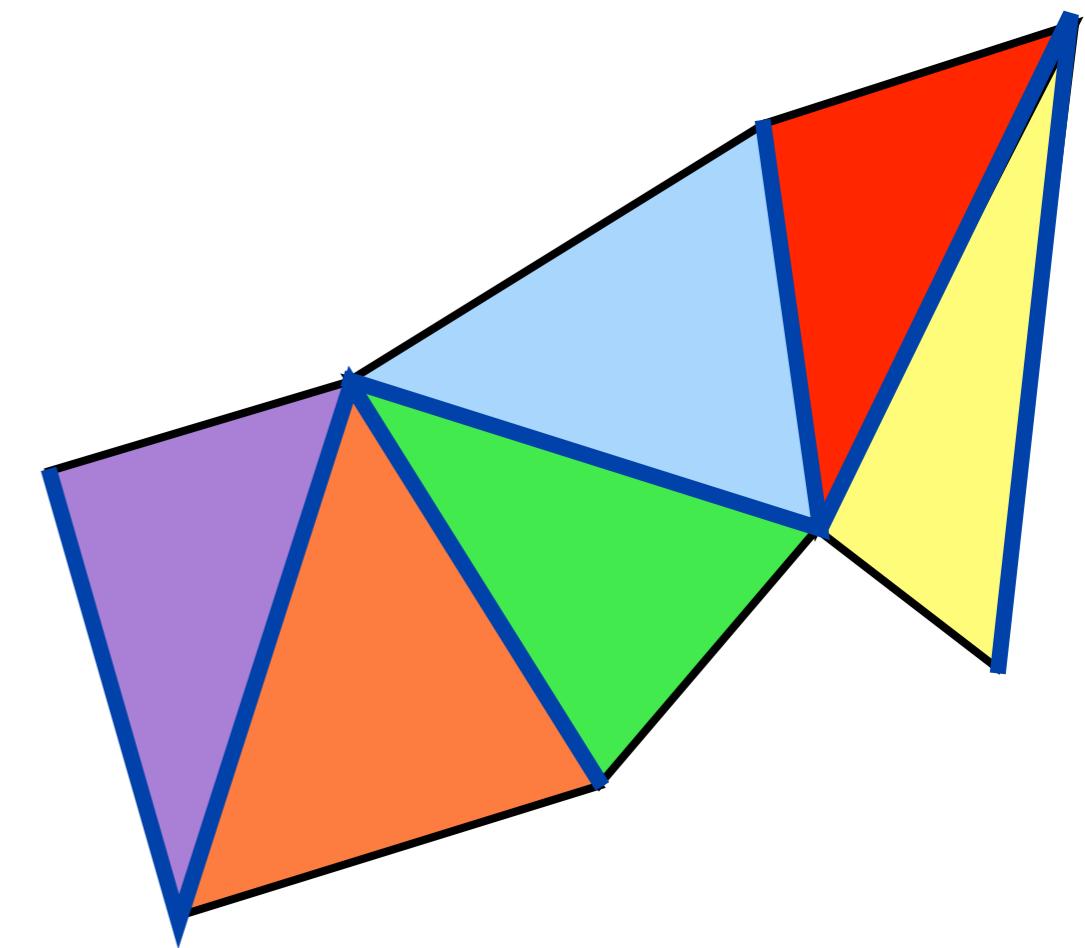


👉 Polygons

# Better than polygons



👉 triangle fans



👉 triangle strips

# Image based modeling



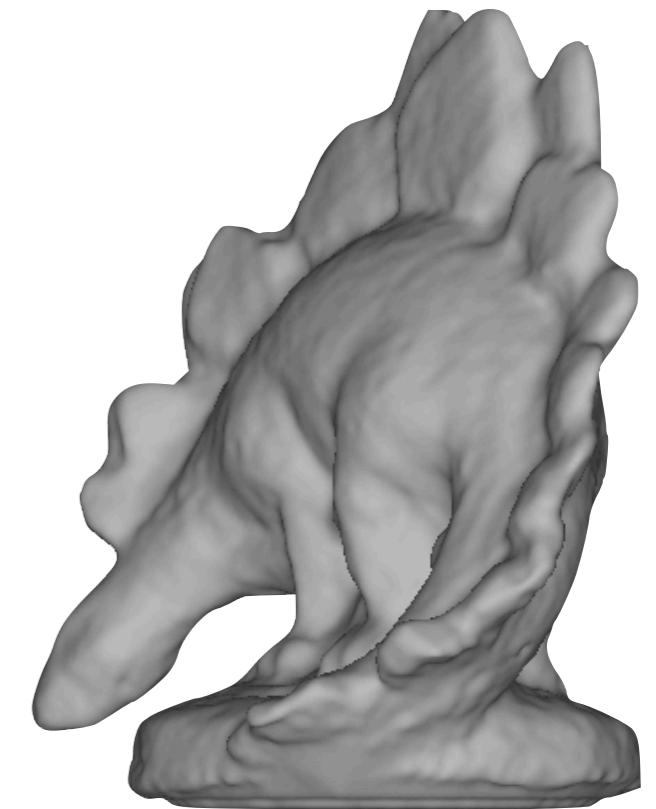
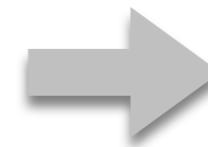
Camera mapping ([blenderguru.com](http://blenderguru.com))

# Image based modeling



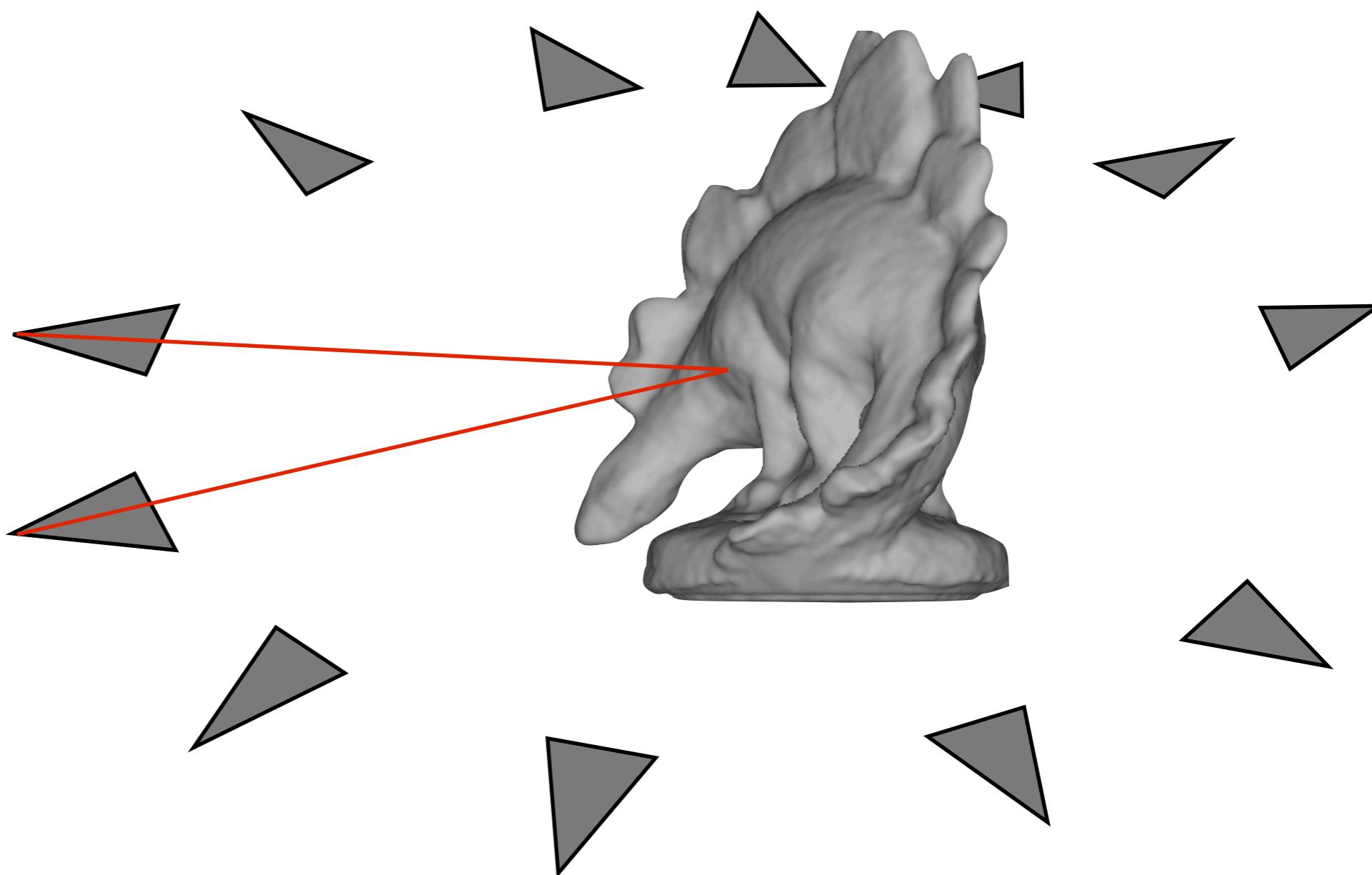
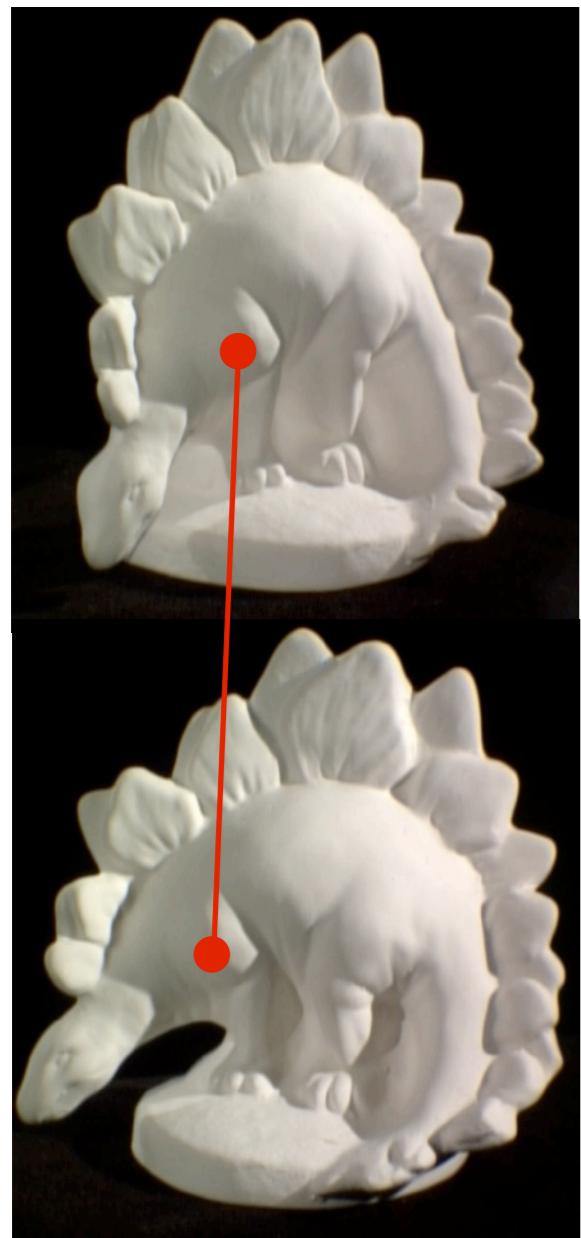
Camera mapping ([blenderguru.com](http://blenderguru.com))

# Image-based modeling



multiview reconstruction

# Image-based modeling



👉 **multiview reconstruction**

# Procedural modeling

