

RENDU

- IG3D -

Jonathan Fabrizio

jo.fabrizio.free.fr

www.lrde.epita.fr/~jonathan

LRDE - EPITA

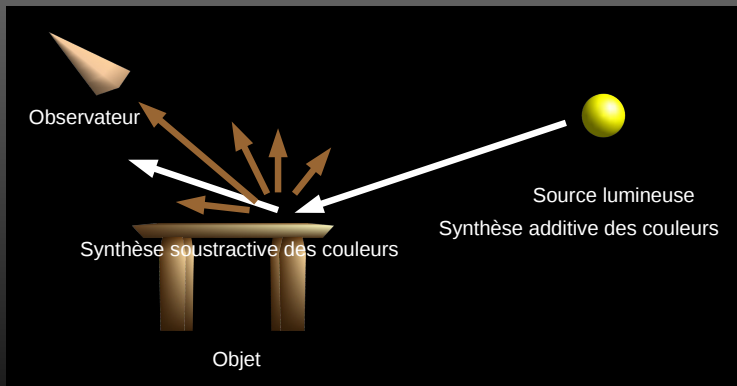
2017



Rendu temps réel vs Rendu photoréaliste

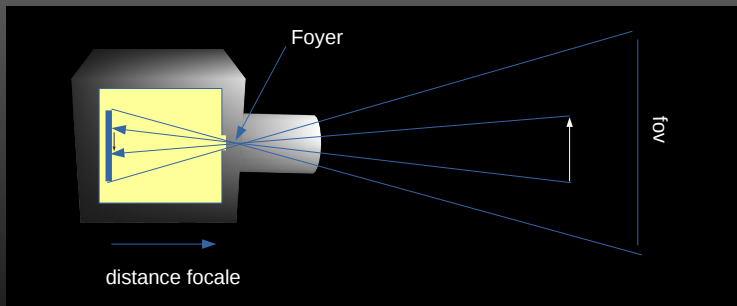
- ▶ Rendu photoréaliste
 - ▶ Objectif :
 - ▶ Génération d'images réalistes
 - ▶ Contrainte de temps faible...
 - ▶ Stratégies :
 - ▶ Object-based rendering algorithms
Illumination globale calculée indépendamment du point de vue
 - ▶ Image-based rendering algorithms
Illumination calculée partiellement, en fonction du point de vue
 - ▶ Deterministic rendering algorithms
 - ▶ Monte Carlo rendering algorithms
- ▶ Rendu temps réel
 - ▶ Objectif
 - ▶ Génération rapide d'images

Formation de l'image



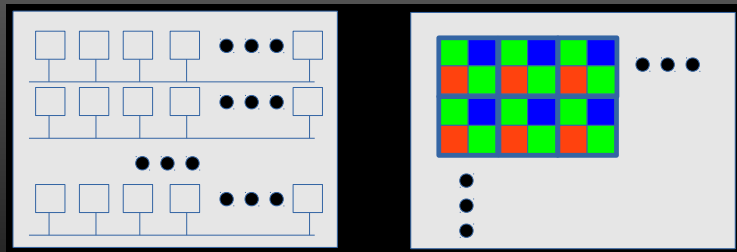
Formation de l'image

Capture de l'image



Formation de l'image

Capture de l'image (capteur CCD, CMOS)



Rendu photoréaliste

- ▶ Raytracing
- ▶ PathTracing
- ▶ PBGI
- ▶ ...

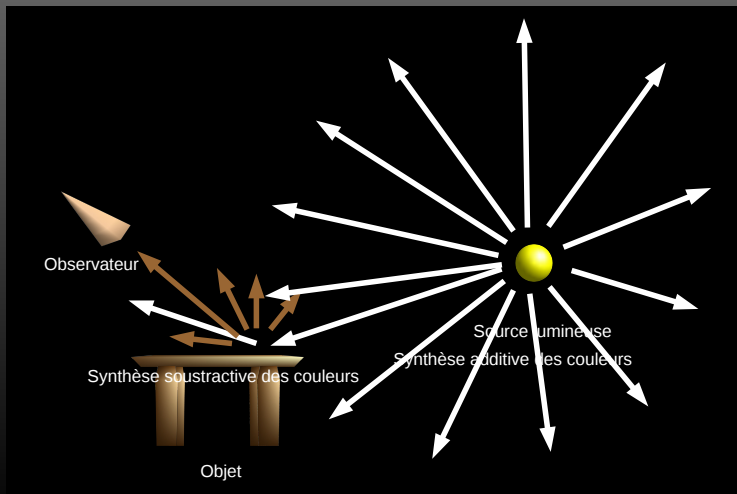
- ▶ Raiosité
- ▶ Photonmap
- ▶ ...

Rendu photoréaliste

- ▶ On peut facilement modéliser la caméra
- ▶ Il faut réussir à modéliser l'éclairage
 - ▶ Idée : «suivre» les rayons lumineux pour trouver le chemin parcouru depuis la source lumineuse, jusqu'à l'œil.

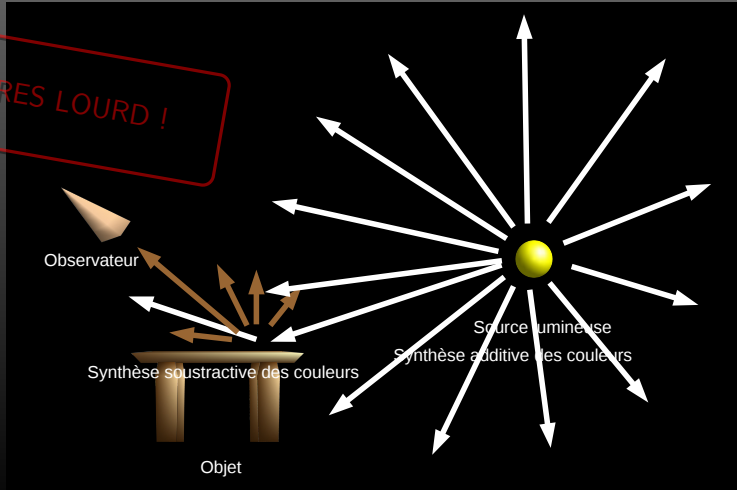
Rendu photoréaliste

- ▶ Principe : Lancer une "infinité" de rayons depuis la source pour espérer trouver ceux qui frappent l'œil de l'observateur



Rendu photoréaliste

- ▶ Principe : Lancer une "infinité" de rayons depuis la source pour espérer trouver ceux qui frappent l'œil de l'observateur

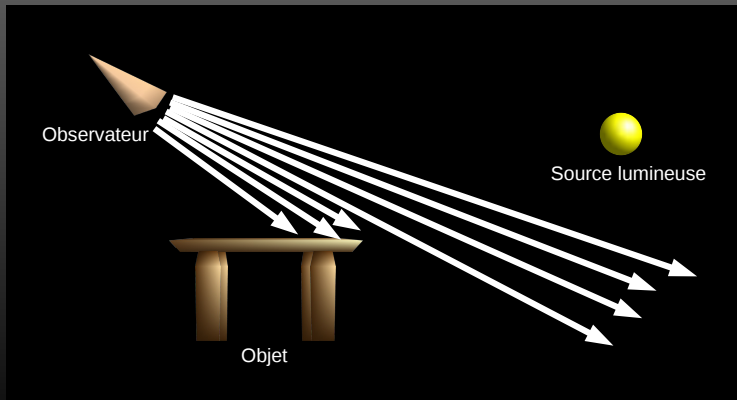


Raytracing

- ▶ Historique
 - ▶ 68, Appel (du raycasting ??),
 - ▶ 80, Whitted (ajoute les effets optiques : reflexion, transparence...).
- ▶ Principe
 - ▶ Idée de base : Difficile de suivre tous les rayons partant de la source en revanche il est possible d'estimer le chemin inverse.

Raytracing

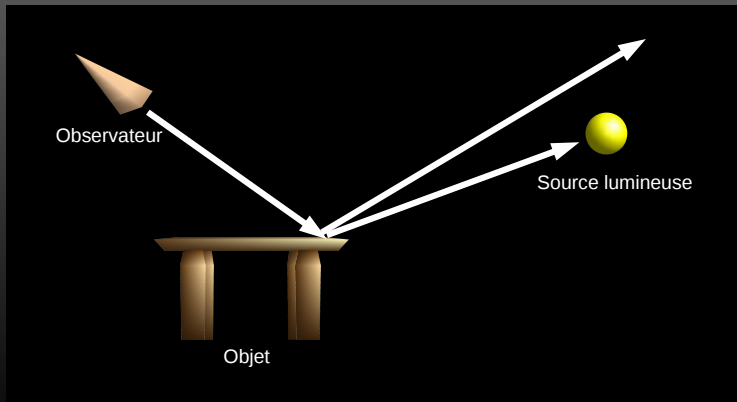
- ▶ Principe
 - ▶ Faire le chemin inverse pour trouver les objets «vus»



Raytracing

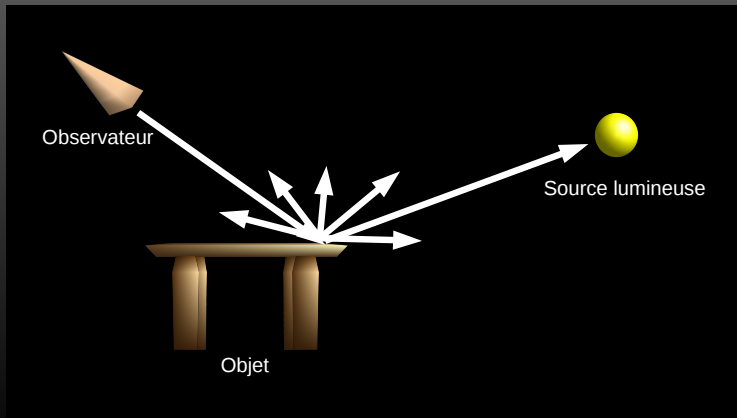
► Principe

- Pour chaque objet vu, on peut estimer une approximation de l'éclairage local



Raytracing

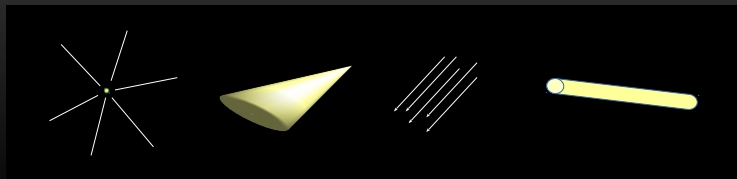
- ▶ Principe :
 - ▶ Approximation de deux types de contributions :
 - ▶ la partie diffuse
 - ▶ la partie spéculaire



- ▶ Calcul de l'illumination locale :
 - ▶ Composante diffuse
 - ▶ Composante spéculaire
 - ▶ Apport des sources primaires
 - ▶ Apport des sources secondaires

Raytracing

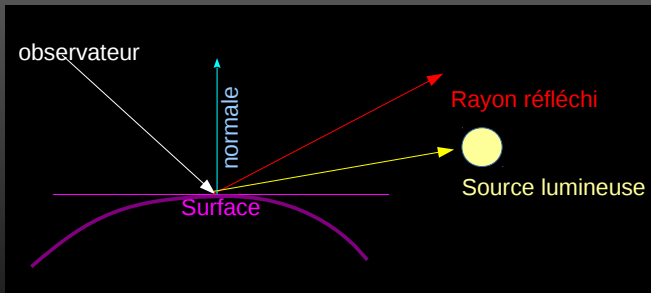
- ▶ Calcul de l'illumination locale :
 - ▶ Composante diffuse
 - ▶ Composante spéculaire
 - ▶ Apport des sources primaires
 - ▶ Apport des sources secondaires
- ▶ Source primaires :
 - ▶ Lumières ponctuelles
 - ▶ Spots
 - ▶ Lumières directionnelles
 - ▶ Objets lumineux



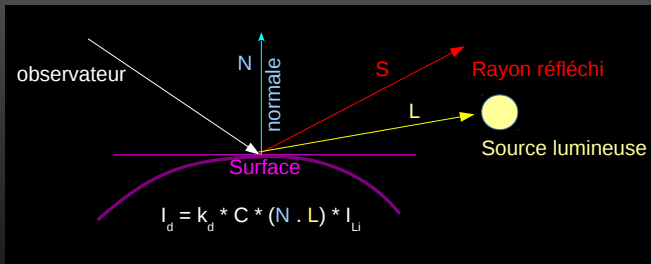
- ▶ Calcul de l'illumination locale :
 - ▶ Composante diffuse
 - ▶ Composante spéculaire
 - ▶ Apport des sources primaires
 - ▶ Apport des sources secondaires

- ▶ Source secondaires :
 - ▶ Les autres objets éclairés

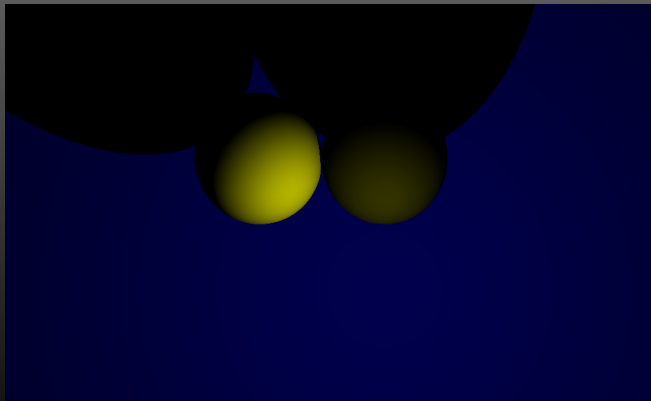
► Modèle local



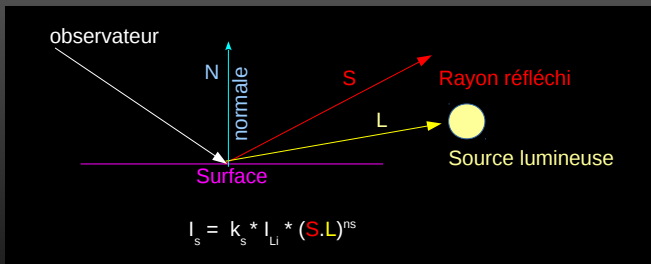
- ▶ La composante diffuse
 - ▶ La propriété de diffusion de la surface est k_d
 - ▶ La couleur de la surface est C



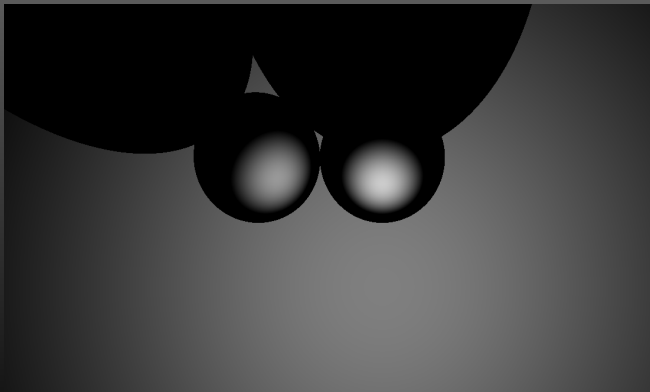
- ▶ La componente diffuse



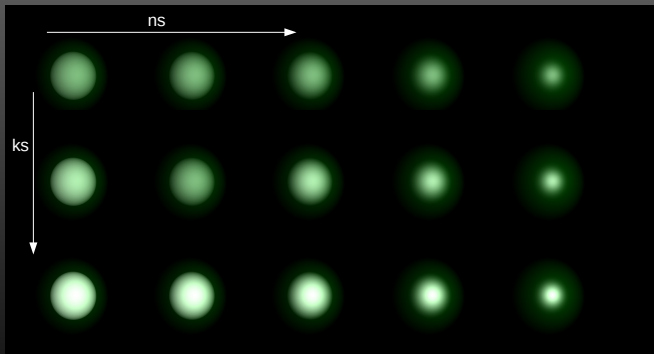
- ▶ La composante spéculaire
 - ▶ La propriété de réflexion de la surface est k_s



- ▶ La composante spéculaire

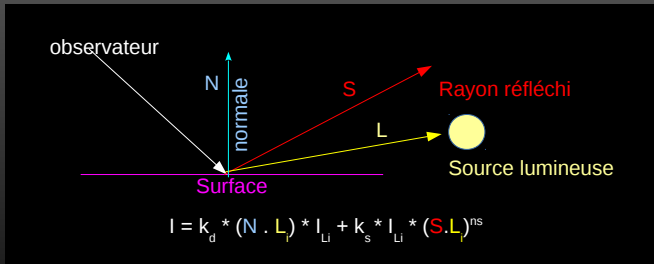


► La composante spéculaire



Raytracing

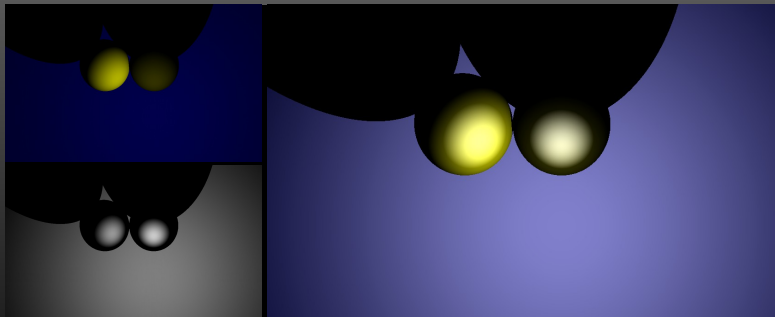
- ▶ Calcul de l'illumination locale :
 - ▶ Composante diffuse
 - ▶ Composante spéculaire
 - ▶ Apport des sources primaires
 - ▶ Ne tient pas compte des sources secondaires



- ▶ Les « k_d » incluent la couleur
- ▶ Il faut sommer toutes les sources lumineuses $i...$

Raytracing

Résultat :



Étape 1 : Prise en compte des sources primaires

Algorithme :

- ▶ Pour l'ensemble des points de l'image :
 - ▶ Calculer le vecteur directeur du rayon lumineux v partant de l'observateur
 - ▶ Chercher les intersections de ce rayon lumineux avec l'intégralité des objets de la scène et garder le plus proche
 - ▶ Calculer le niveau d'éclairement au point d'intersection en sommant l'apport diffus et spéculaire pour chaque source lumineuse

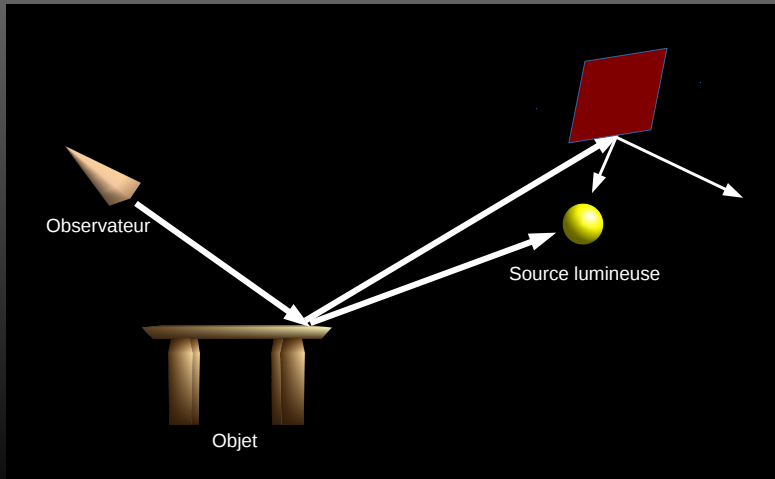
Étape 1 : Prise en compte des sources primaires

Problèmes :

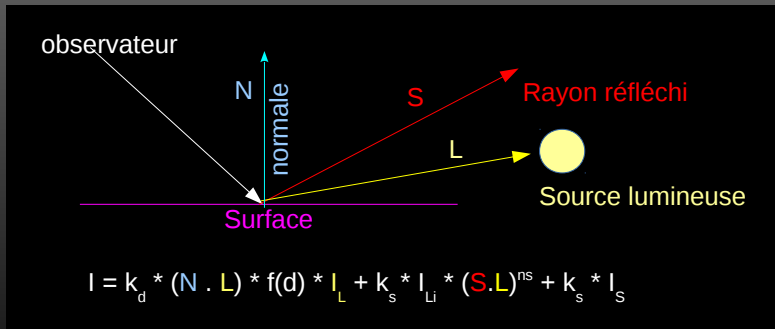
- ▶ Ne tient pas compte des sources lumineuses secondaires
- ▶ Ne gère pas les ombres

Raytracing

Prise en compte des sources secondaires :



Prise en compte des sources secondaires :



Étape 2 : Prise en compte des sources primaires et certaines sources secondaires

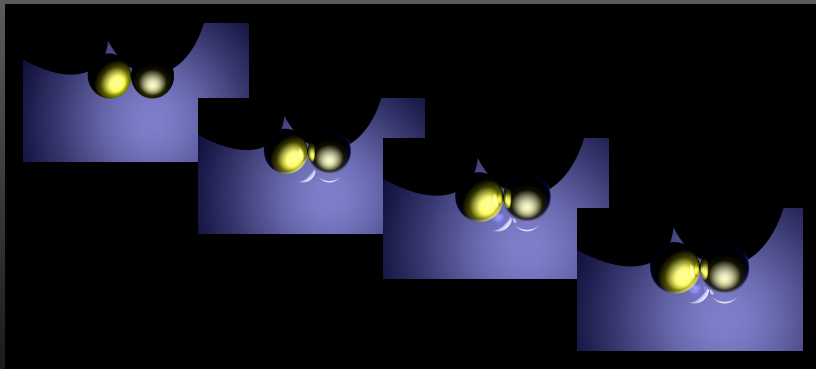
Algorithme :

- ▶ Pour l'ensemble des points de l'image :
 - ▶ Calculer le vecteur directeur du rayon lumineux v partant de l'observateur
 - ▶ Chercher les intersections de ce rayon lumineux avec l'intégralité des objets de la scène et garder le plus proche
 - ▶ Relancer un rayon dans la direction de S puis calculer le niveau d'éclairément récursivement
 - ▶ Calculer le niveau d'éclairément au point d'intersection en sommant l'apport diffus et spéculaire pour chaque source lumineuse ainsi que l'éclairément dans la direction de S

Étape 3 : Prise en compte de l'ombre

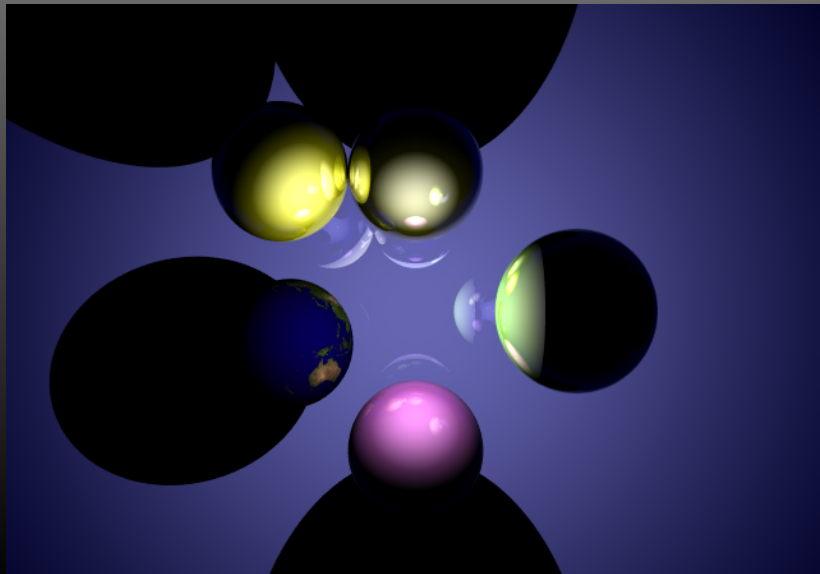
- ▶ Pour l'ensemble des rayons que l'on « lance » vers les sources primaires, il faut chercher si un objet de la scène ne s'est pas inséré entre le point considéré et la source. Pour cela, il faut à nouveau calculer l'intersection du rayon avec l'ensemble des objets de la scène et prendre le plus proche.

Résultats :



Raytracing

Résultats :



Raytracing

- ▶ L'algorithme du raytracing est un processus simple, récursif
- ▶ Il faut être capable, pour chaque objet, de calculer la normale en chaque point
- ▶ Il faut réfléchir à la condition d'arrêt

Raytracing

- ▶ Avantages
- ▶ Inconvénients

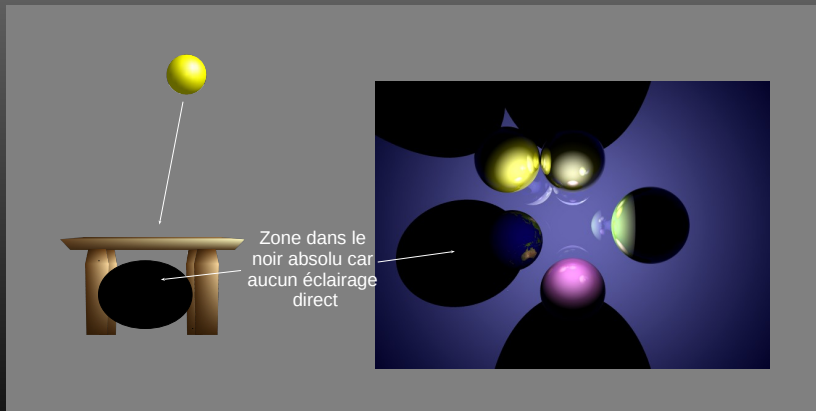
- ▶ Avantages
 - ▶ Algorithme simple et rapide à mettre en œuvre
 - ▶ Génère des images honorables
 - ▶ ...
- ▶ Inconvénients
 - ▶ Temps de calcul un peu élevé
 - ▶ Pas gestion de la profondeur de champ et autres effets
 - ▶ Mauvaise gestion des ombres (frontières trop brutales)
 - ▶ Sources secondaire pas suffisamment prises en compte (éclairage indirect incorrect)
 - ▶ Objets transparents
 - ▶ « Aliasing »
 - ▶ ...

Raytracing

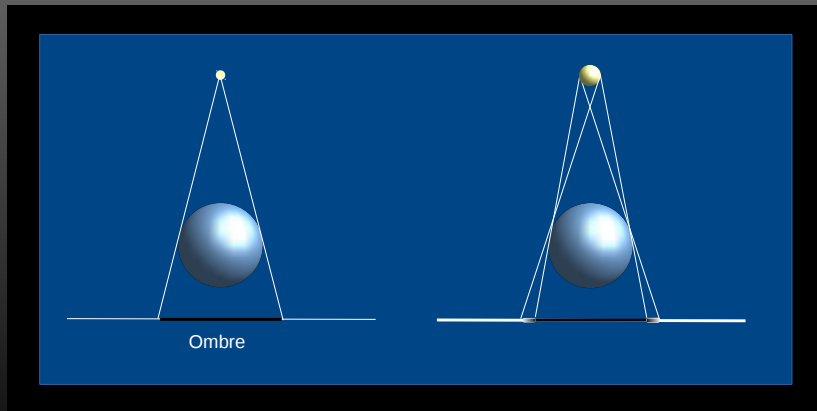
Problème de l'aliasing



Problème de de l'éclairage indirect



Problème de l'ombre



Conclusion

- ▶ Algorithme simple
- ▶ Nécessite beaucoup d'améliorations pour avoir des images photoréalistes.