

# Insertion d'objets virtuels



GIF-4105/7105 Photographie Algorithmique  
Jean-François Lalonde

Merci à A. Efros et P. Debevec!

# Plan — reste de la session

- Cette semaine: insertion d'objets virtuels
- Autres sujets: à votre choix!
  - “Big Data”: comprendre et synthétiser les images à partir de grandes quantités de données visuelles
  - Comment prendre de bonnes photos?
  - Comment fonctionne la Kinect?
  - Caméras “lightfield”: focaliser après avoir pris la photo!
    - [lytro.com](http://lytro.com)
  - Détecter les fausses images
  - ACP et visages
  - Caméras algorithmiques
  - Autres idées?

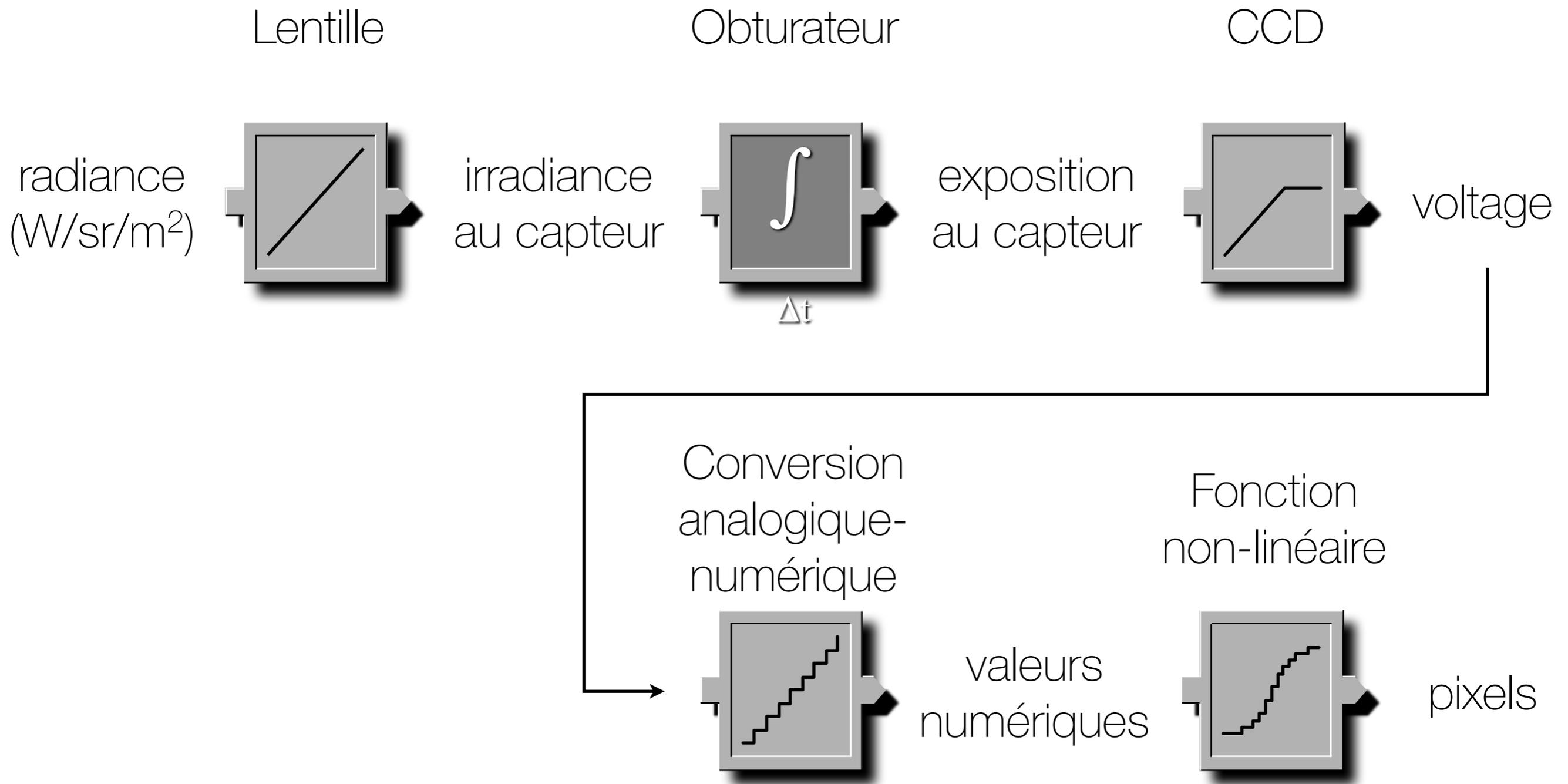


**Allez voter sur le groupe  
Facebook!**

# Aujourd'hui

- Retour sur la formation d'images à haute plage dynamique
- Reproduction tonale ("tone mapping")
- Insertion d'objets virtuels
  - À conclure mercredi

# Modèle radiométrique (simplifié)



# Modèle radiométrique (plus simplifié)

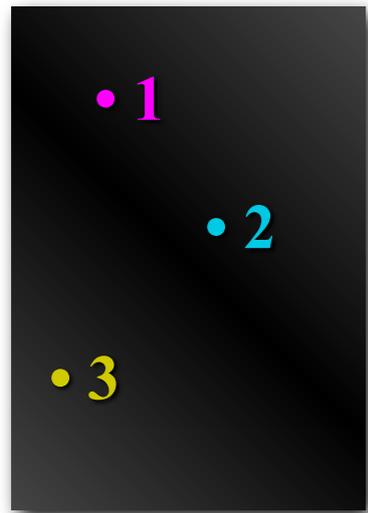


$$z = f(\text{exposition})$$

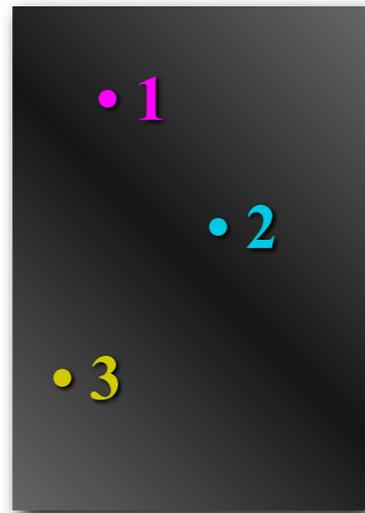
$$\text{exposition} = \text{radiance} \times \Delta t$$

# Algorithme

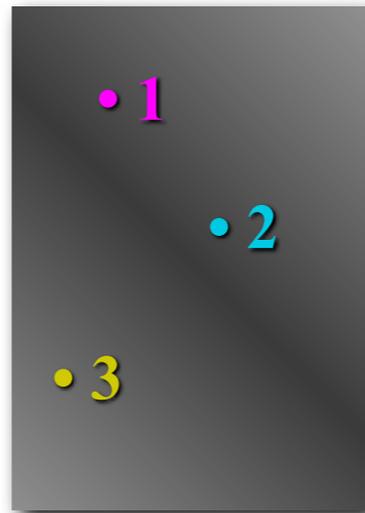
Série d'images



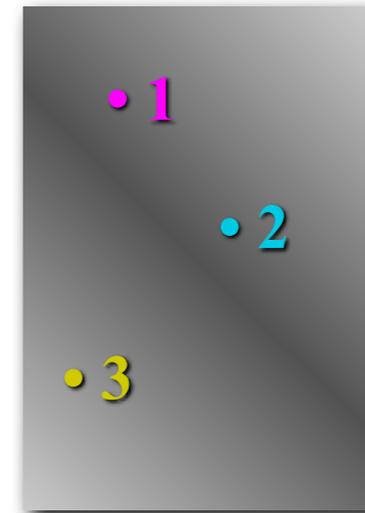
$\Delta t = 1/64 \text{ sec}$



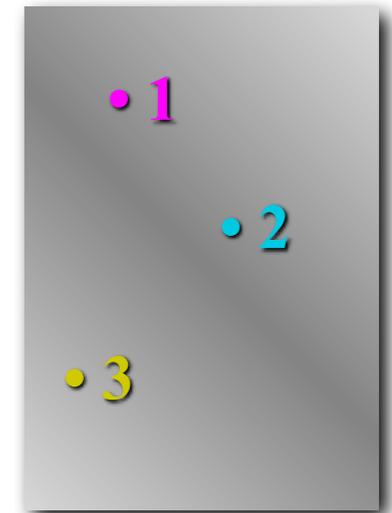
$\Delta t = 1/16 \text{ sec}$



$\Delta t = 1/4 \text{ sec}$



$\Delta t = 1 \text{ sec}$



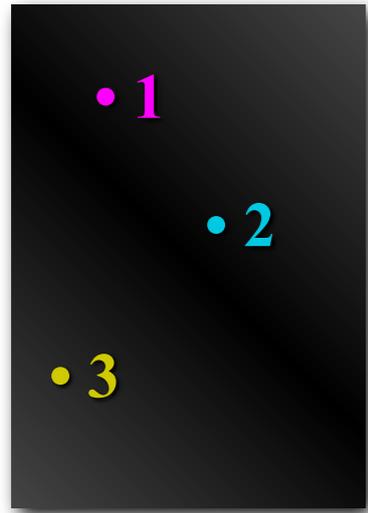
$\Delta t = 4 \text{ sec}$

$$z = f(\text{exposition})$$

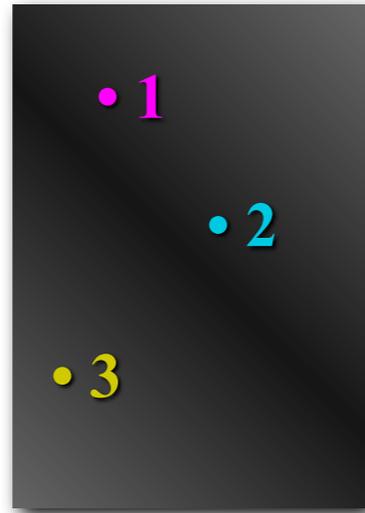
$$\text{exposition} = \text{radiance} \times \Delta t$$

# Algorithme

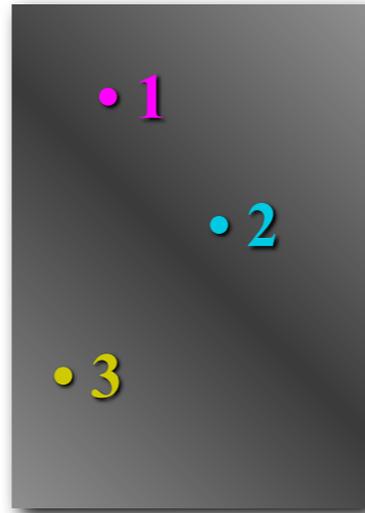
Série d'images



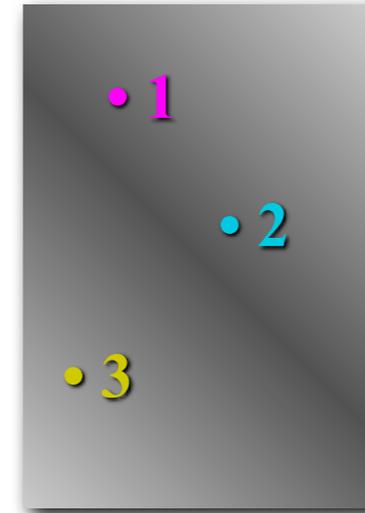
$\Delta t = 1/64$  sec



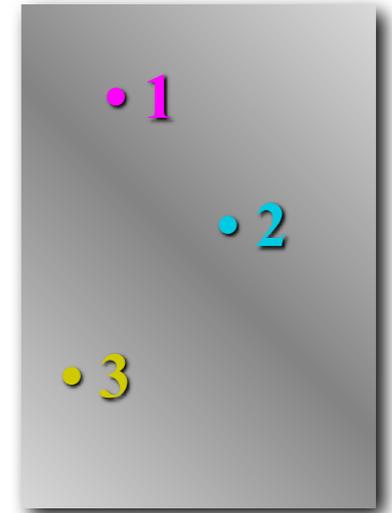
$\Delta t = 1/16$  sec



$\Delta t = 1/4$  sec



$\Delta t = 1$  sec



$\Delta t = 4$  sec

$$z_{ij} = f(\text{exposition}_{ij})$$

$$\text{exposition}_{ij} = \text{radiance}_i \times \Delta t_j$$

$$z_{ij} = f(\text{radiance}_i \times \Delta t_j)$$

$$\text{radiance}_i \times \Delta t_j = f^{-1}(z_{ij})$$

$$\log \text{radiance}_i + \log \Delta t_j = g(z_{ij})$$

# Math

- Notons la fonction inverse discrétisée:  $g(z)$
- Pour chaque pixel  $i$  dans une image  $j$ , nous avons:

$$\log \text{radiance}_i + \log \Delta t_j = g(z_{ij})$$

$$r_i + t_j = g(z_{ij})$$

- Système d'équations linéaires sur-contraint:

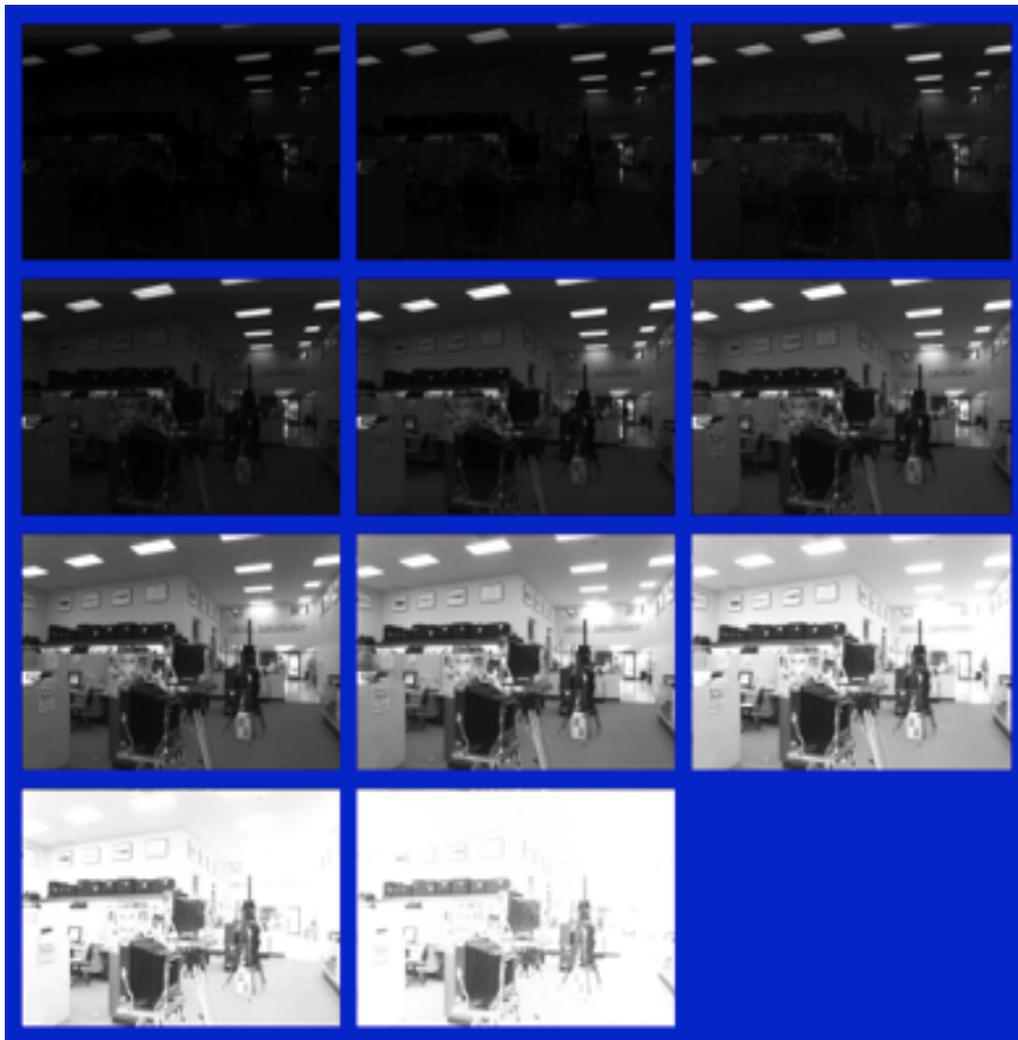
$$\sum_{i \in \text{pixels}} \sum_{j \in \text{images}} (r_i + t_j - g(z_{ij}))^2 + \lambda \sum_{z=0}^{255} g''(z)^2$$

composante pour s'assurer  
qu'on approxime les données

composante s'assurer  
que la courbe soit lisse

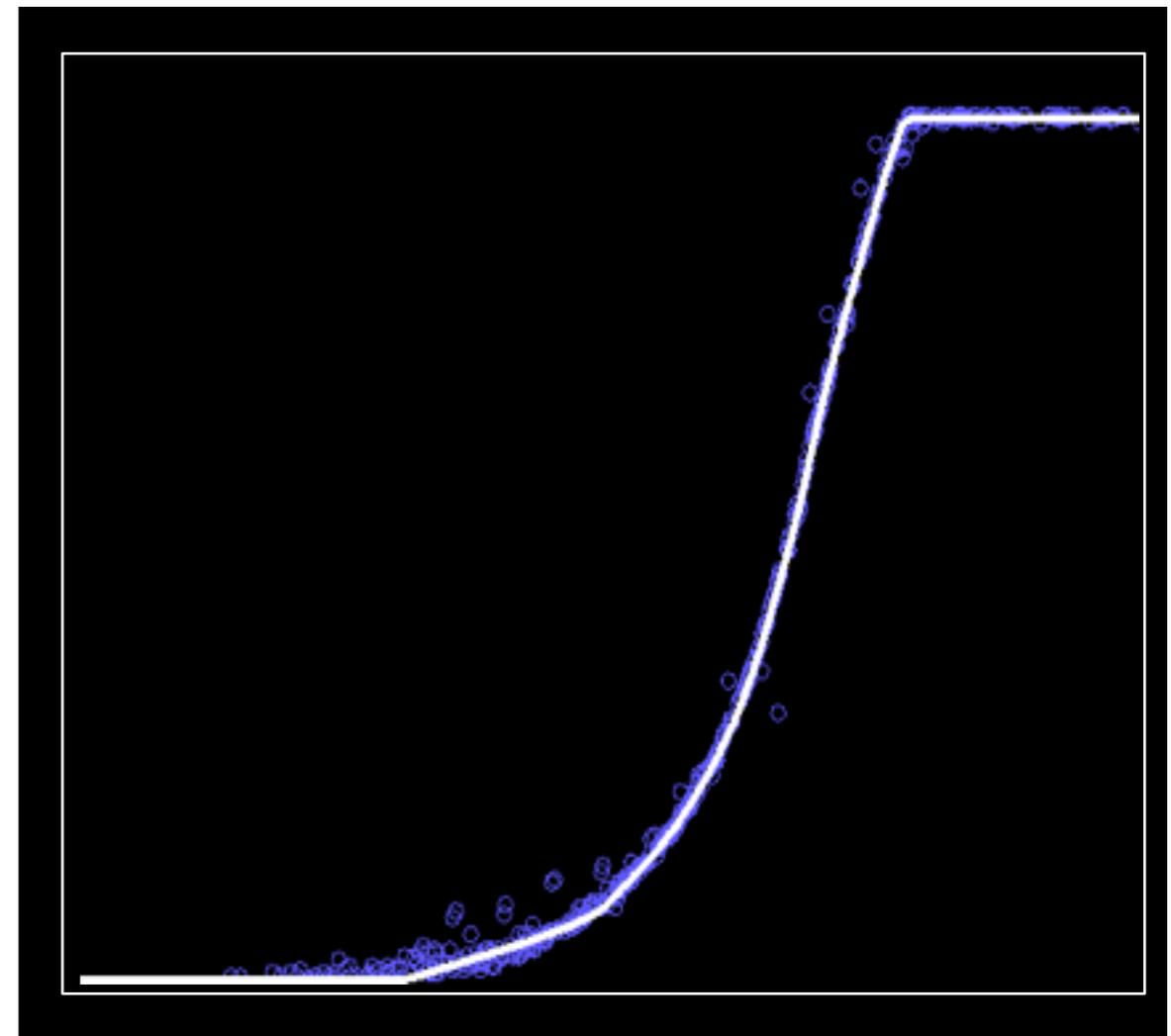
# Résultat

Kodak DCS460  
1/30 à 30 sec



Pixel

Courbe estimée

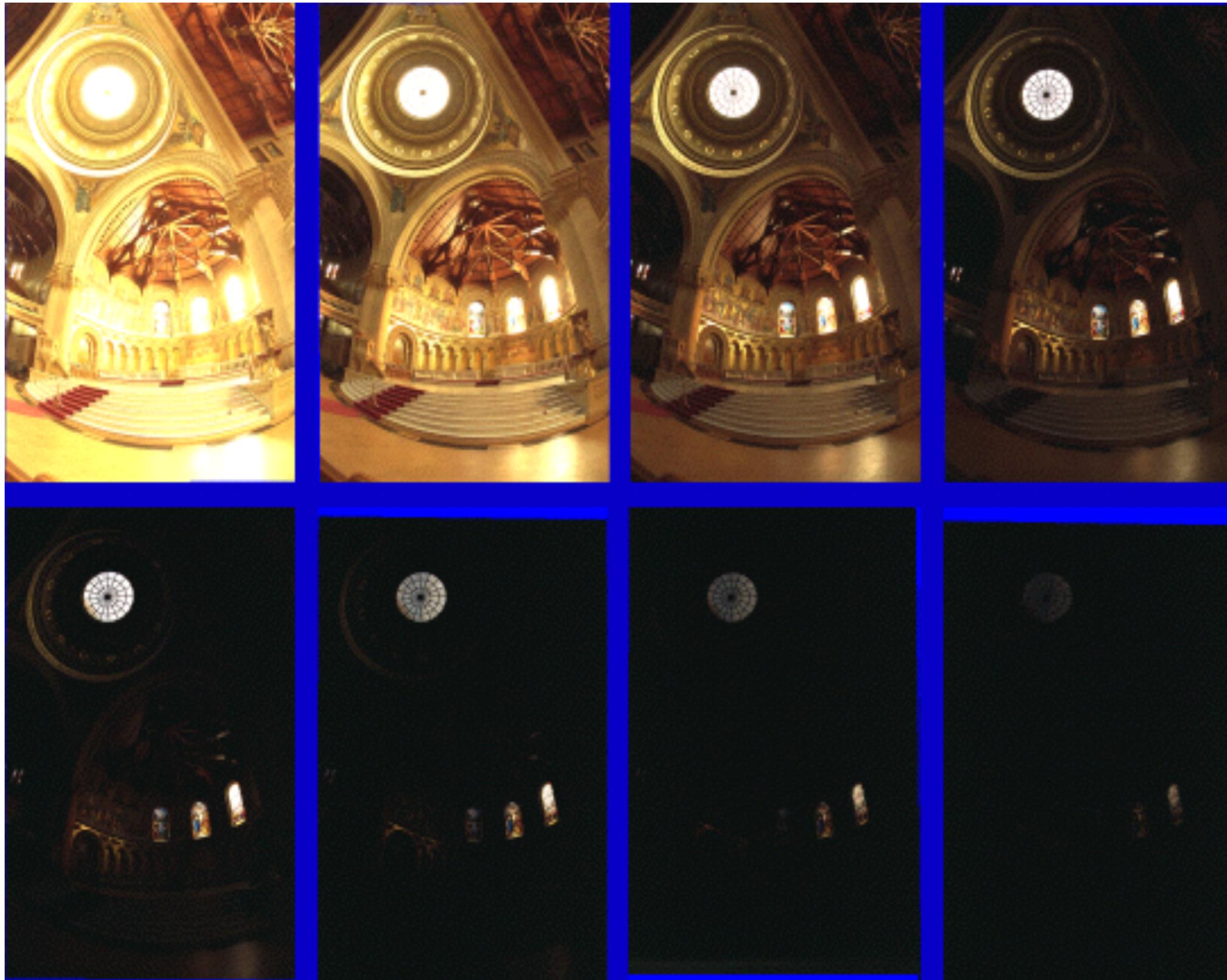


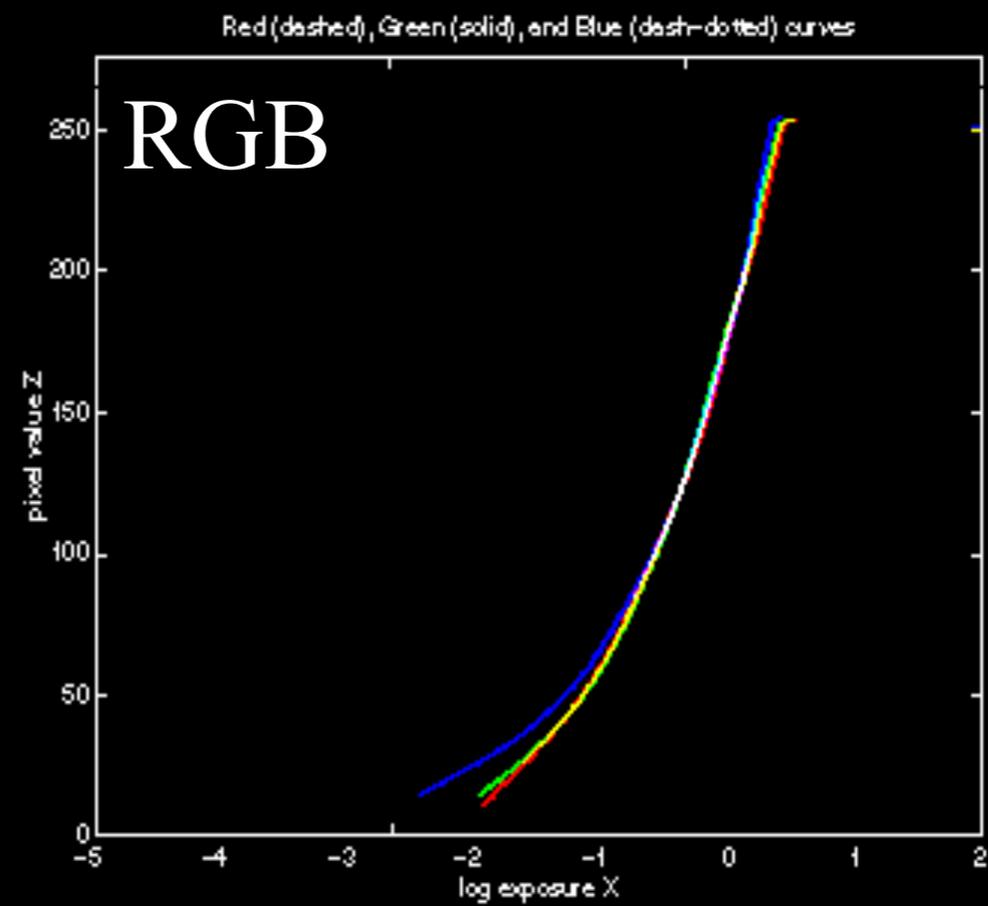
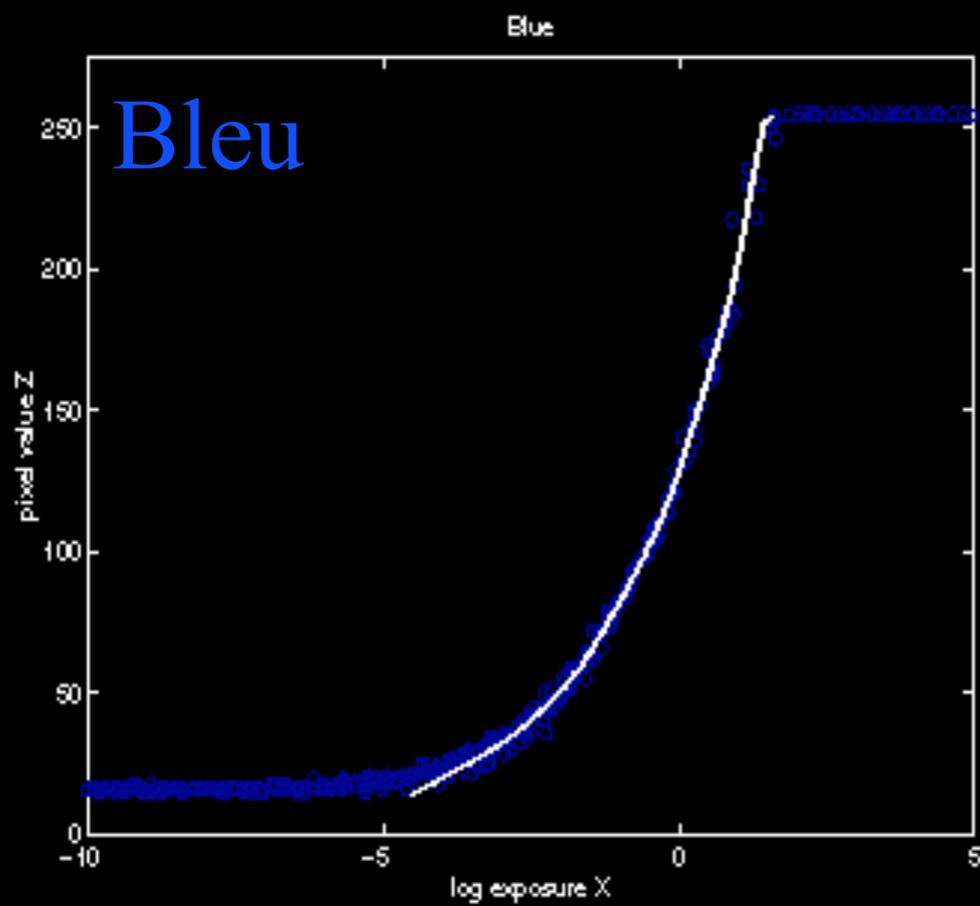
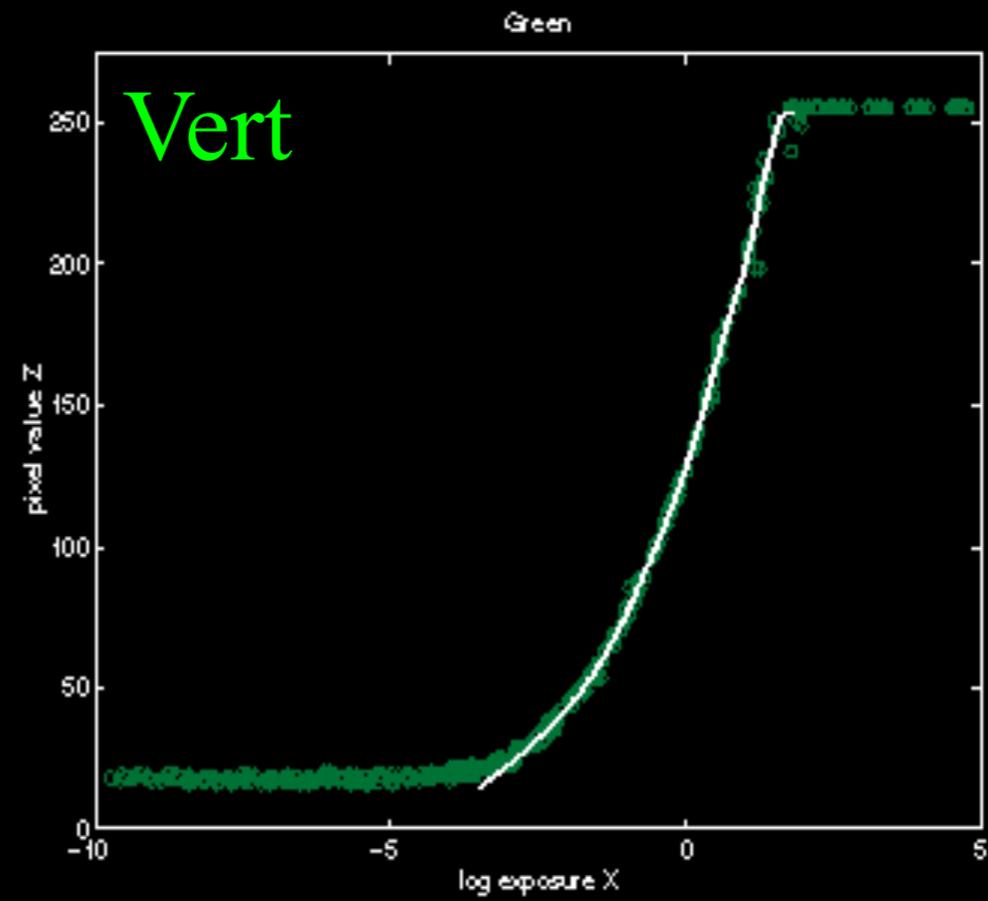
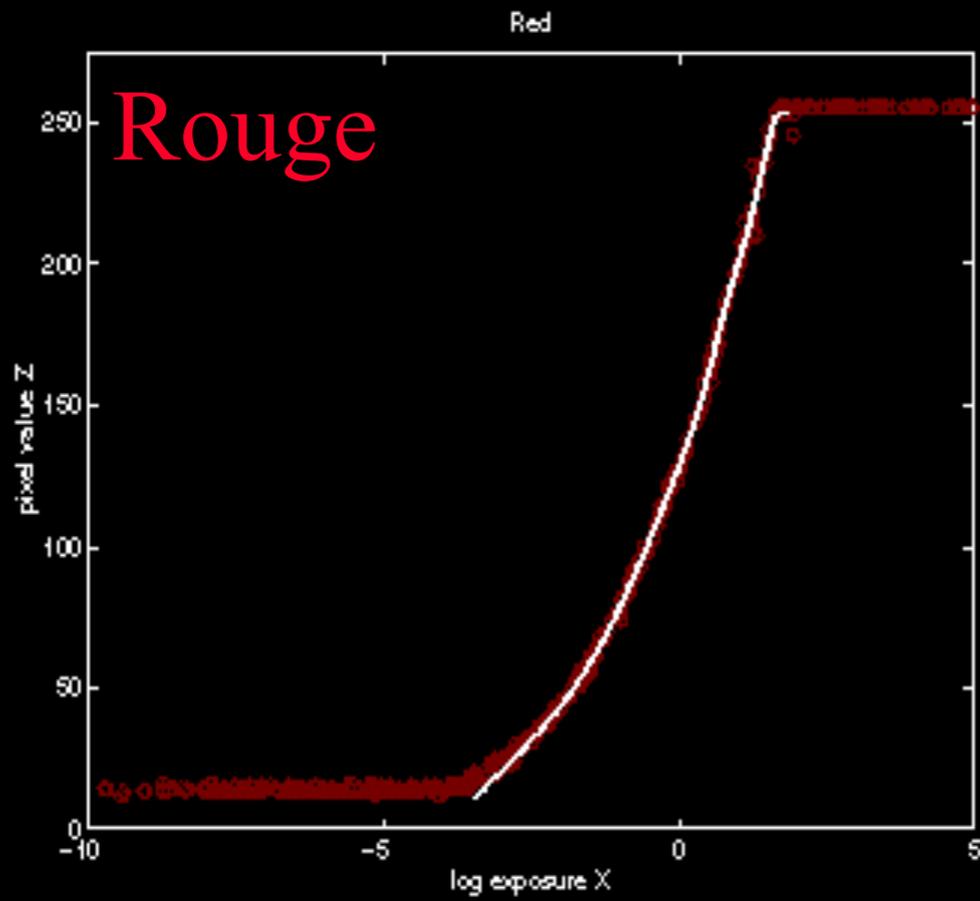
log exposition

# Radiance

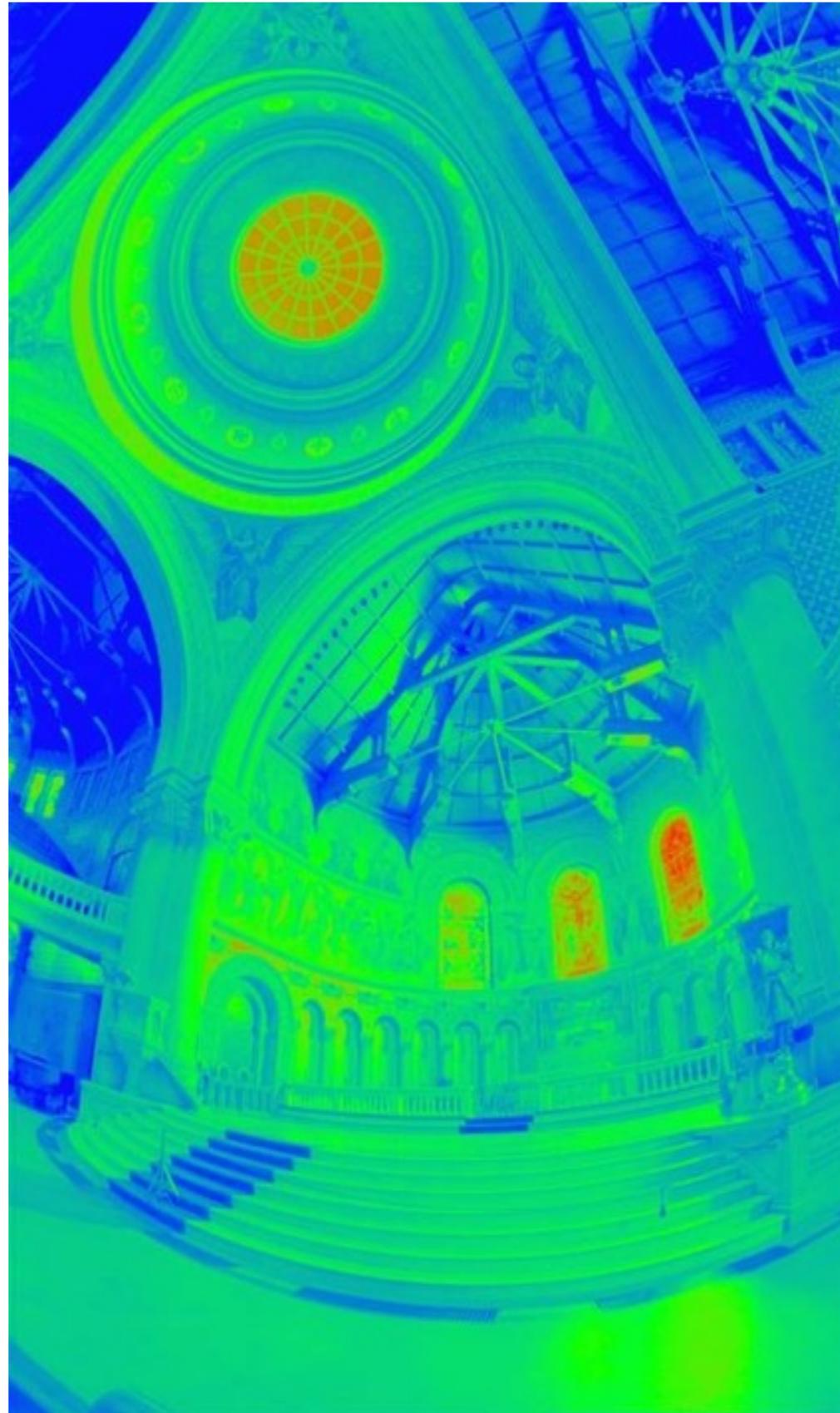


# Résultats: couleur





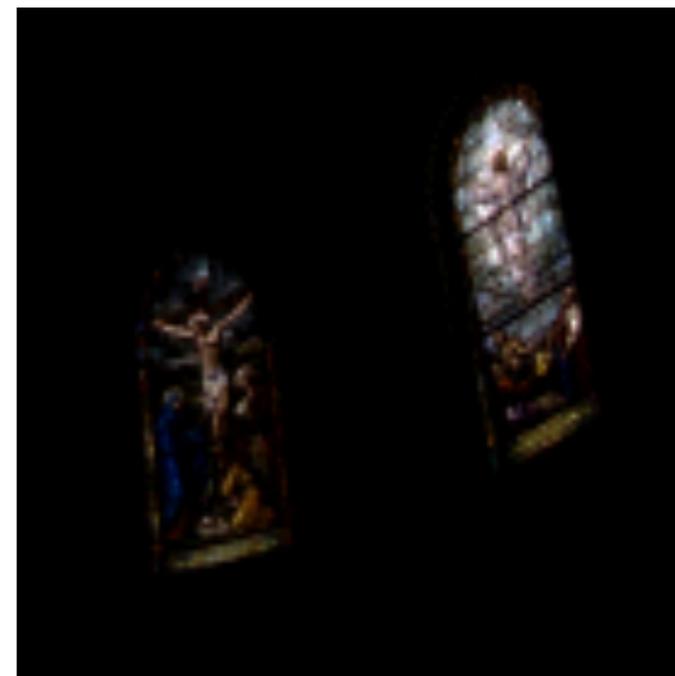
# Radiance



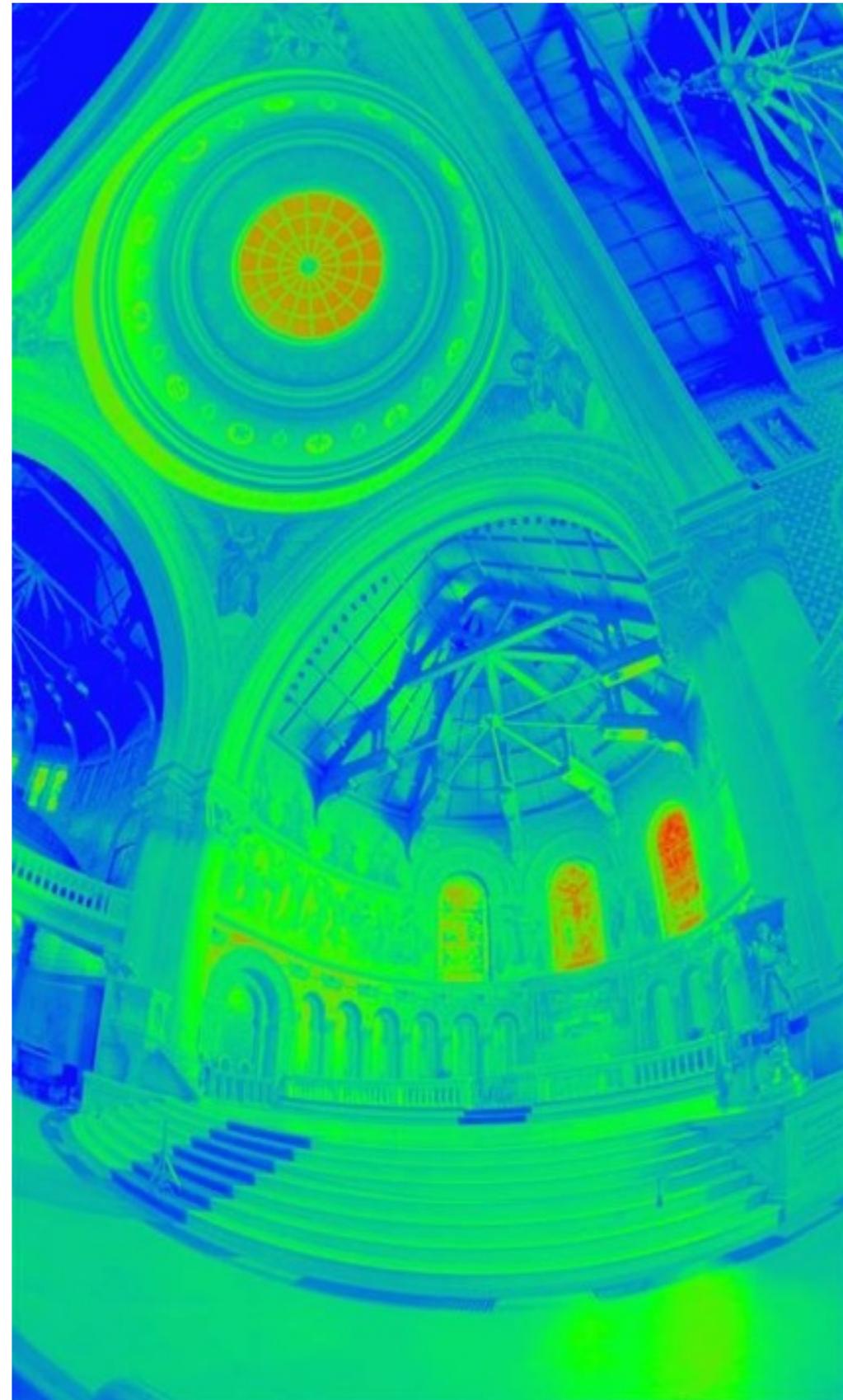
# Radiance



Image précédente  
entre 0 et 255

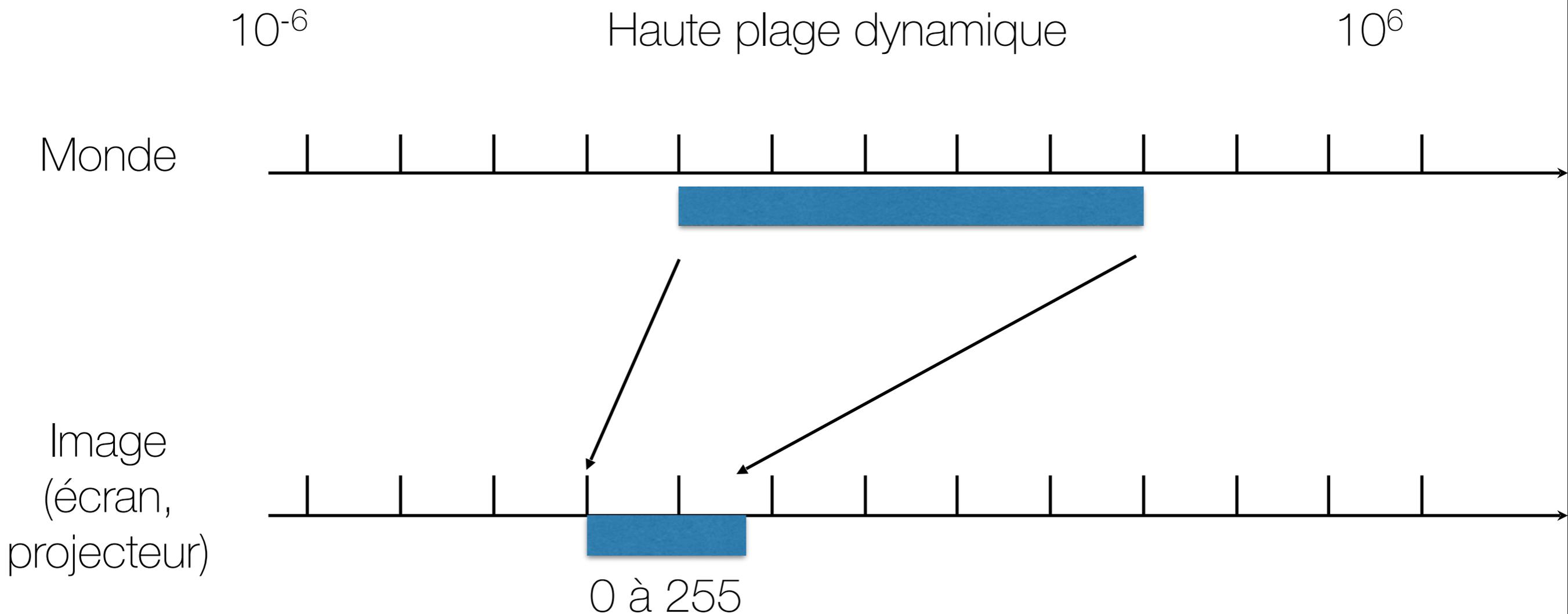


# Et maintenant?



# Reproduction tonale

- Comment faire?
  - Linéaire? Seuil? Suggestions?



# Linéaire



En fonction des pixels  
les plus clairs



En fonction des pixels  
les plus sombres

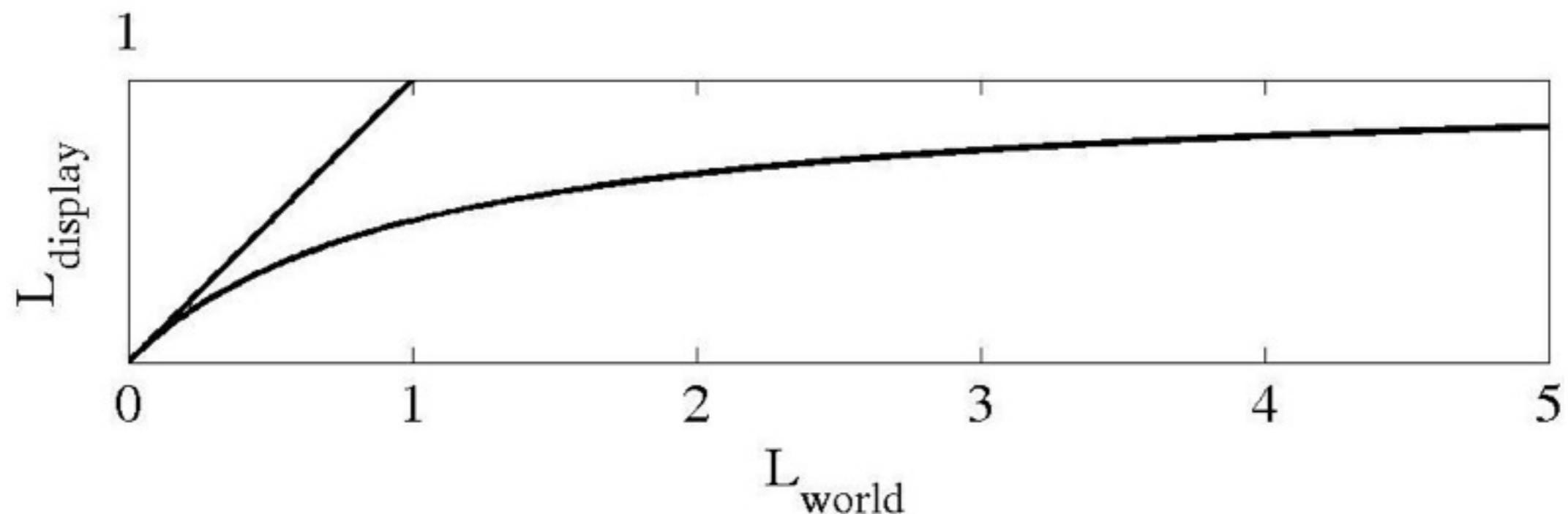
# Opérateur global

- Déterminer une courbe qui:
  - Ramène le contenu du signal HDR dans une plage qui convient à un écran ou un projecteur
  - N'augmente pas les parties sombres
- Donc:
  - Asymptote à 255
  - Dérivée = 1 à 0

# Opérateur global (Reinhard et al.)

- Solution toute simple: utiliser une transformée non-linéaire

$$L_{display} = \frac{L_{world}}{1 + L_{world}}$$



# Non-linéaire



Reinhard

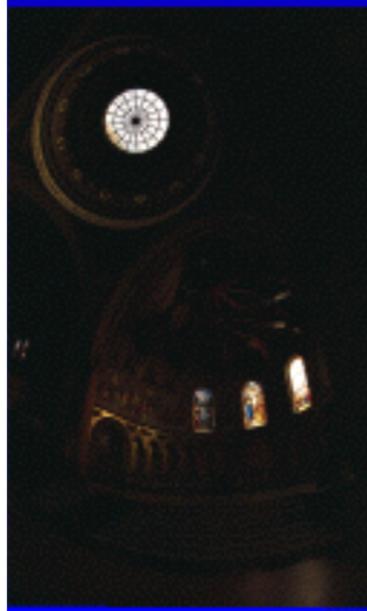
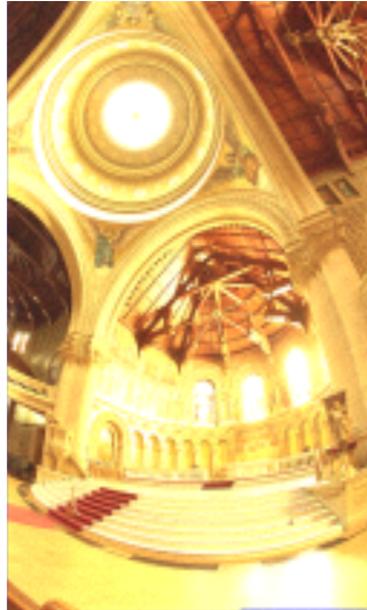


En fonction des pixels  
les plus sombres

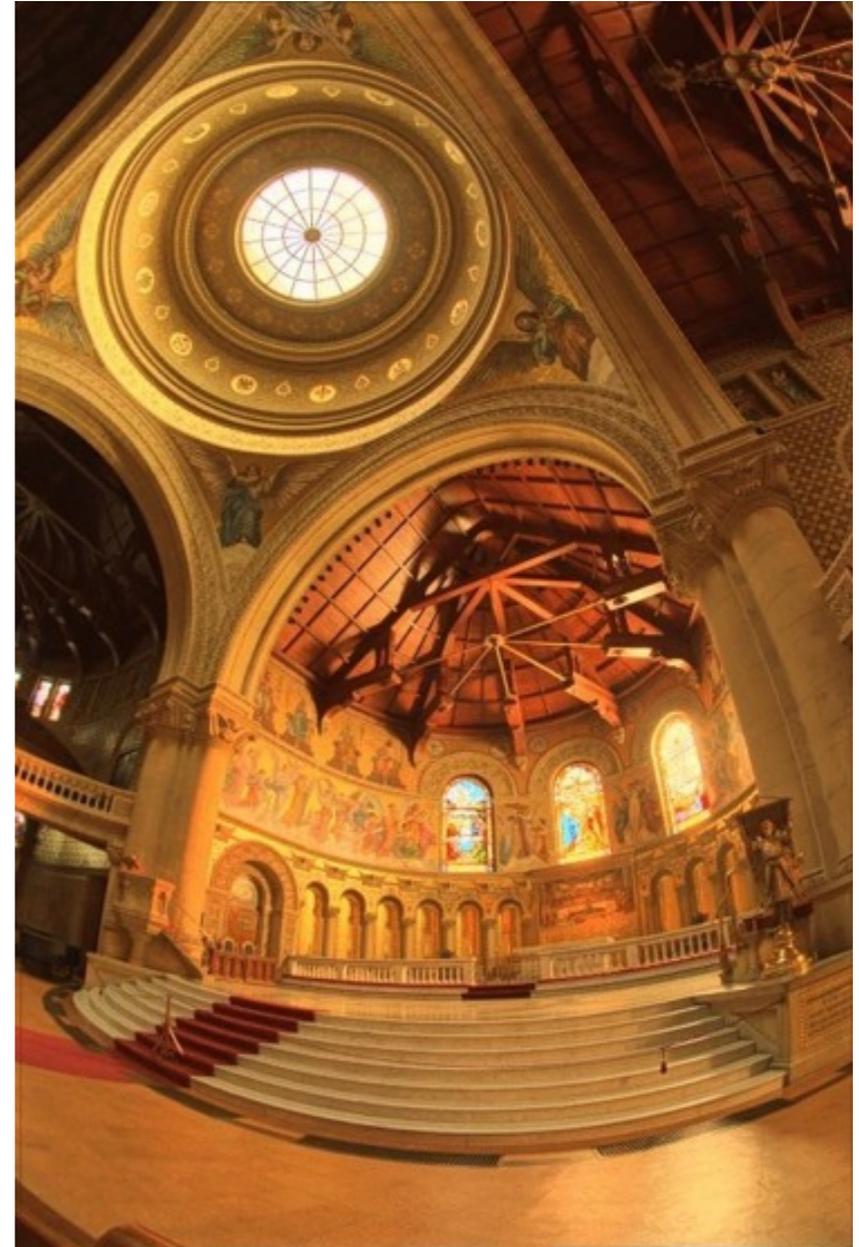
# Opérateur global



# Qu'est-ce que nos yeux voient?



Vs.



# Demo

- LuminanceHDR
  - <http://qtpfsgui.sourceforge.net>

# Insérons un objet virtuel



- Pourquoi le résultat est si mauvais?
  - Mauvaise orientation
  - Mauvais éclairage
  - Pas d'ombre

# Solutions

- Mauvaise orientation
  - Estimer les paramètres de la caméra en fonction de la table.  
Comment?
    - Calibrage géométrique
- Illumination
  - Estimer la position et l'intensité des sources lumineuses à placer dans l'environnement virtuel
- Que faire si l'illumination est complexe?
  - Sources étendues, inter-réflexions, etc...

# Carte d'environnement (environment map)

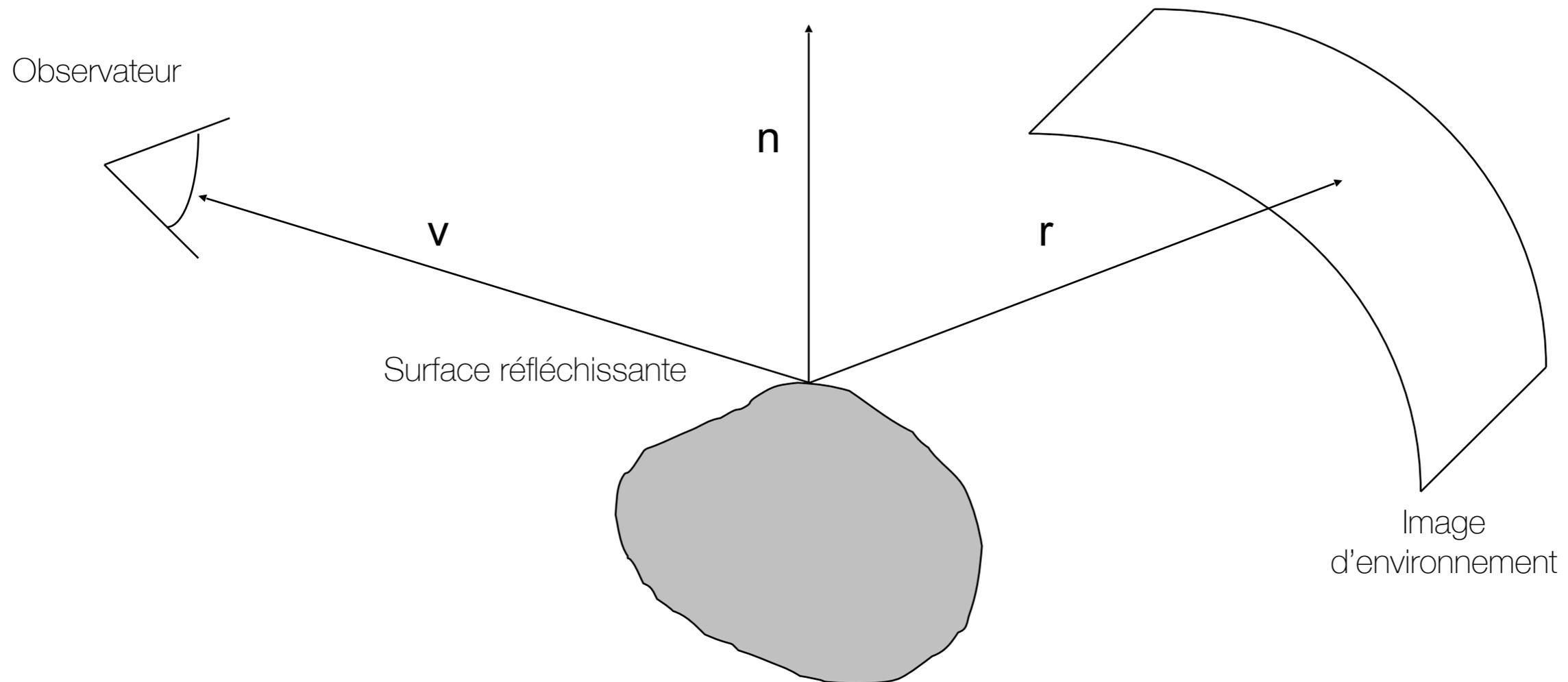
- Solution simple pour objets réfléchissants
  - Modélise l'illumination avec une image panoramique
  - i.e. quantité de radiance qui provient de chaque direction



# Carte d'environnement

Rayon réfléchi:  $r=2(n \cdot v)n-v$

fonction qui convertit le vecteur de réflexion  $(x, y, z)$  en coordonnées image  $(u, v)$



La texture est transférée directement sur l'objet en fonction du vecteur de réflexion et du contenu de l'image d'environnement

# Carte sphérique

- Enregistre la carte en format sphérique ( $\theta$ ,  $\phi$ )
- Pour générer la carte:
  - Effectuer un rendu sphérique à partir de la position de l'objet
- Pour utiliser la carte:
  - Utiliser l'orientation du rayon réfléchi (en coordonnées angulaires), et lire la valeur de couleur correspondantes dans la carte sphérique

# Exemple: carte sphérique



# Exemple

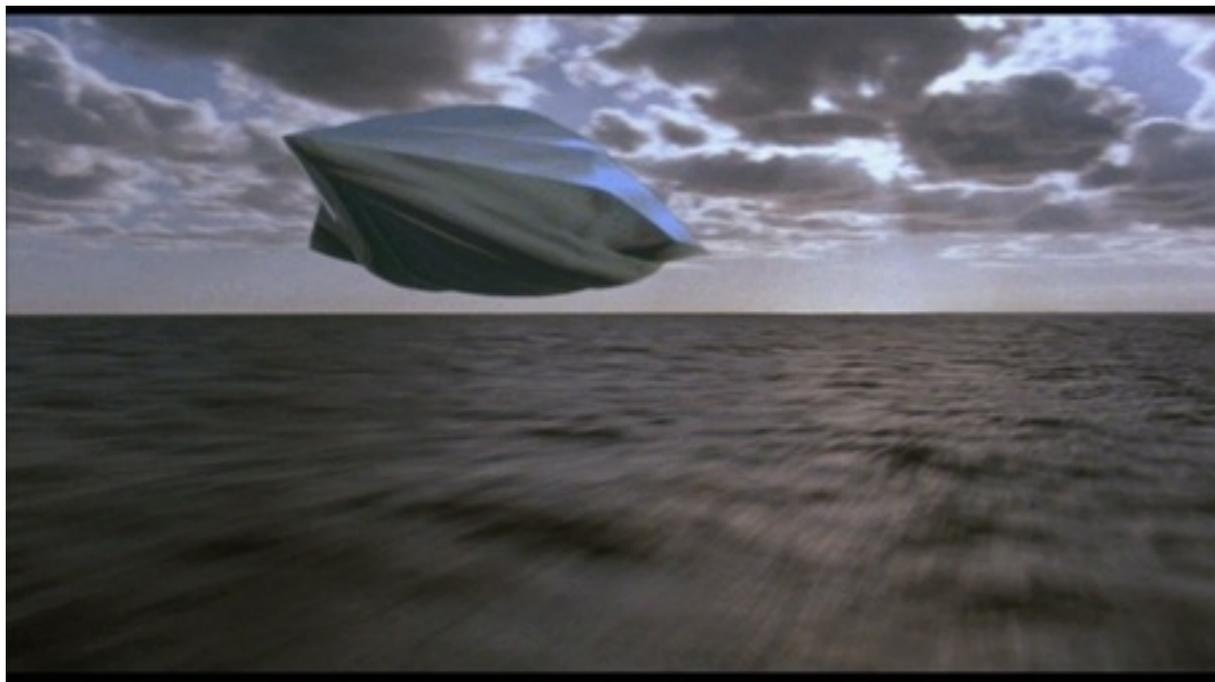
<http://youtu.be/ici8l2foBeQ?t=1m45s>

(source: tchyup sur youtube)

# Approximations

- La carte continent une vue du monde à partir d'un seul point
  - Cette vue change en fonction de la position sur l'objet!
    - Introduit distorsions, mais difficile à remarquer
    - Minimales pour un petit objet dans un grand environnement
- L'objet ne se réfléchit pas lui-même

# Scènes réelles?



Flight of the Navigator (1986), premier film à utiliser cette technologie

# Scènes réelles?



Terminator 2 (1991)

# Carte d'environnement réelles

- Photographier le monde!
- Comment photographier toutes les directions?
  - Panoramas!
  - Existe aussi d'autres solutions...
- Comment photographier les sources lumineuses? Elles sont beaucoup plus lumineuses que le reste de l'environnement...
  - Haute plage dynamique, bien entendu!

# Mosaïques HDR

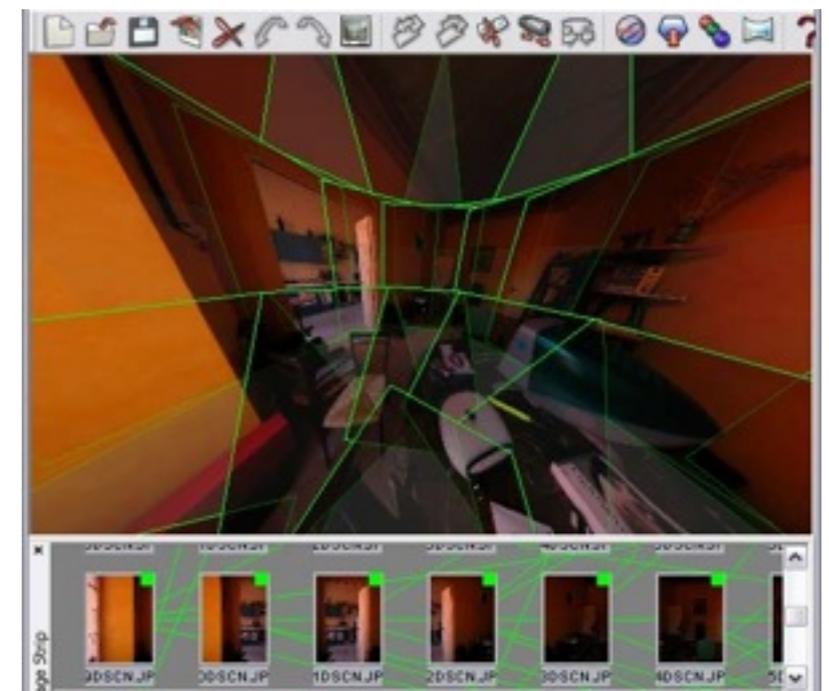
<http://www.gregdowning.com/HDR/stitched/>



**Shadow Detail**



**Highlight Detail**



# Caméras panoramiques

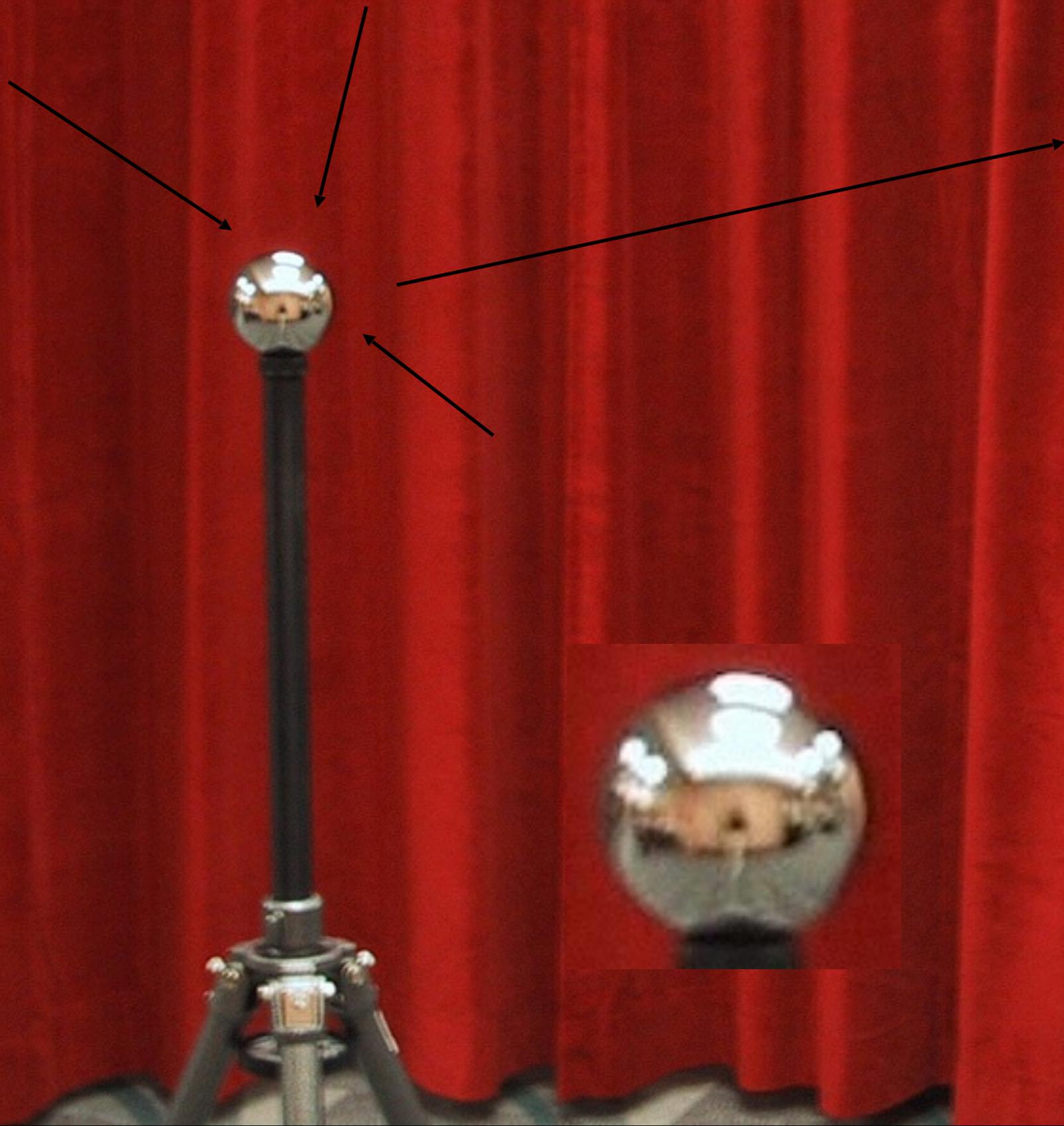
- Avantages:
  - très haute résolution (10K x 7K+)
  - Pas de mosaïques: sphère automatique
  - Bonne plage dynamique (même HDR)
- Problèmes
  - \$\$\$
  - Long à capturer

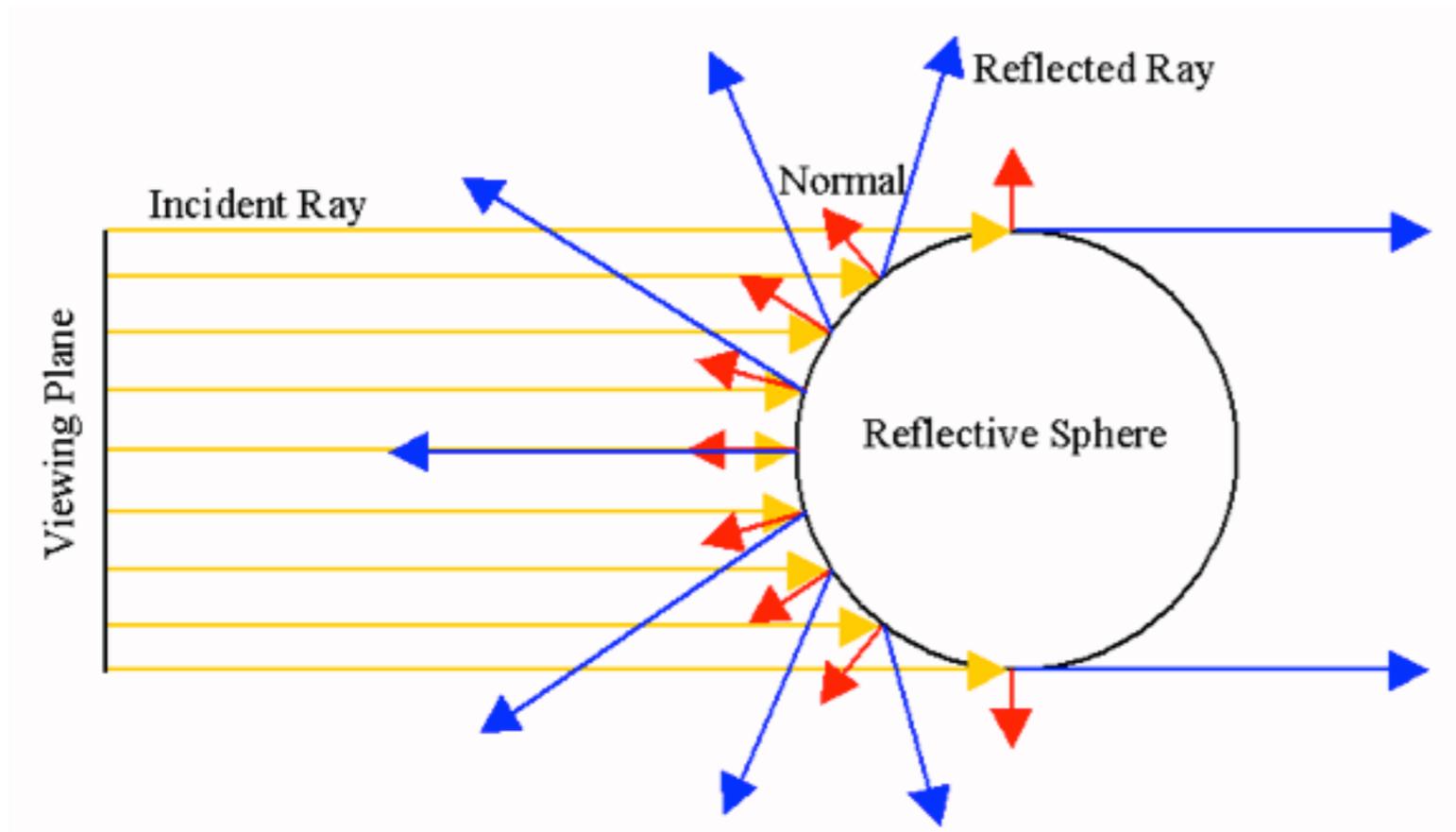


# Lentilles



# Sphère métallique





# Détour: où trouver une sphère métallique?

- 2 pouces de diamètre ~ \$20 ea.
  - McMaster-Carr  
[www.mcmaster.com](http://www.mcmaster.com)
- 6-12 pouces, moins bonne qualité
  - Baker's Lawn Ornaments  
[www.bakerslawnorn.com](http://www.bakerslawnorn.com)
- Hollow Spheres, 2in – 4in
  - Dube Juggling Equipment  
[www.dube.com](http://www.dube.com)



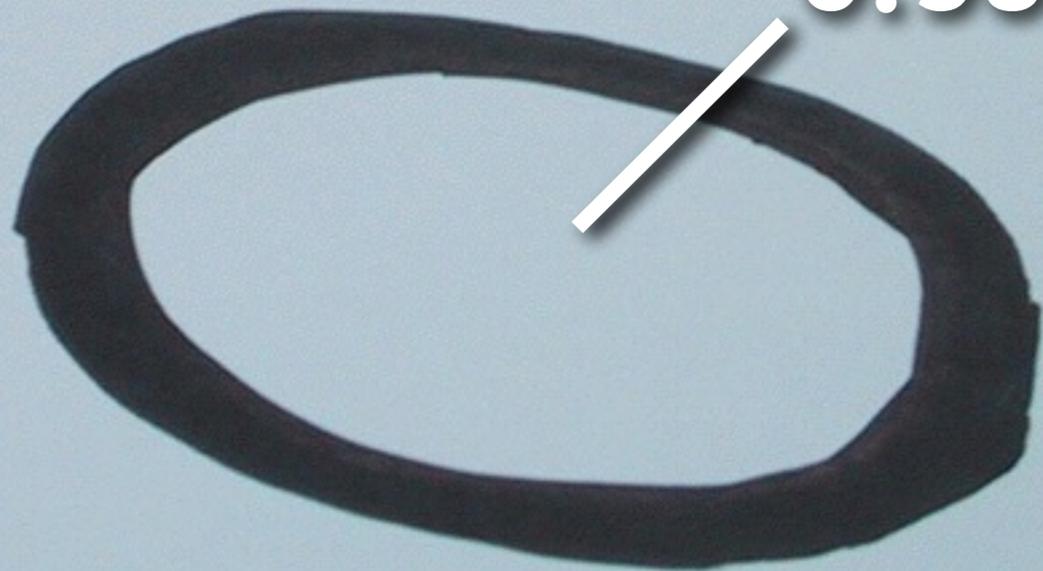


0.34

=> 59%  
réflective

Calibrage de la  
réflexivité de la  
sphère

0.58



Funston  
Beach



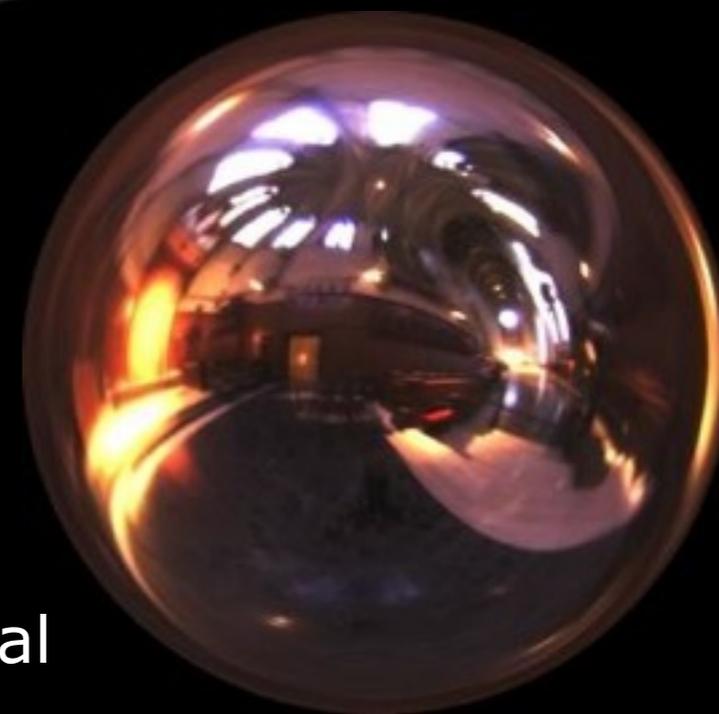
Eucalyptus  
Grove



Uffizi  
Gallery



Grace  
Cathedral

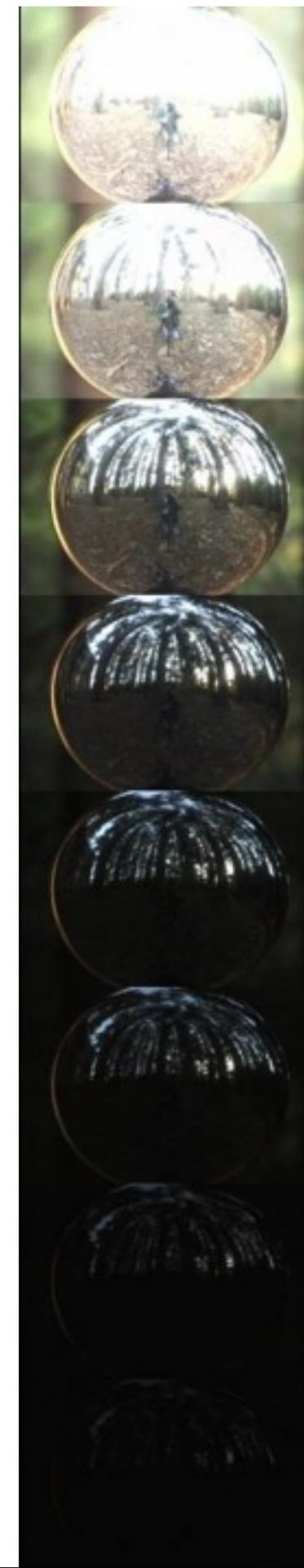
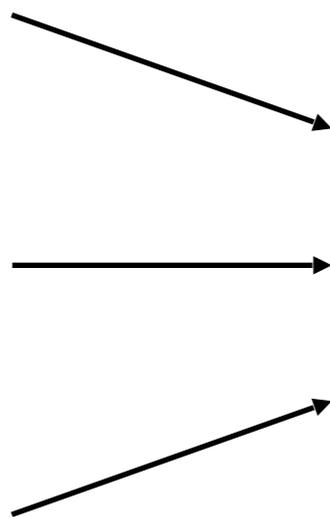
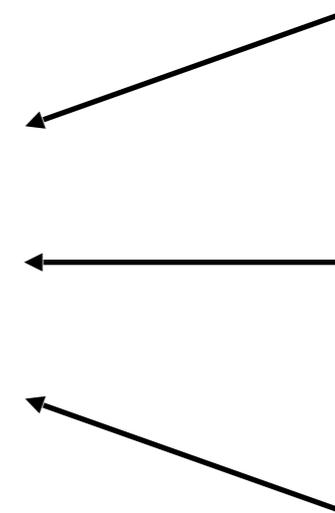


Examples:  
<http://www.debevec.org/Probes/>

Avant

Arrière

# Capturer l'environnement

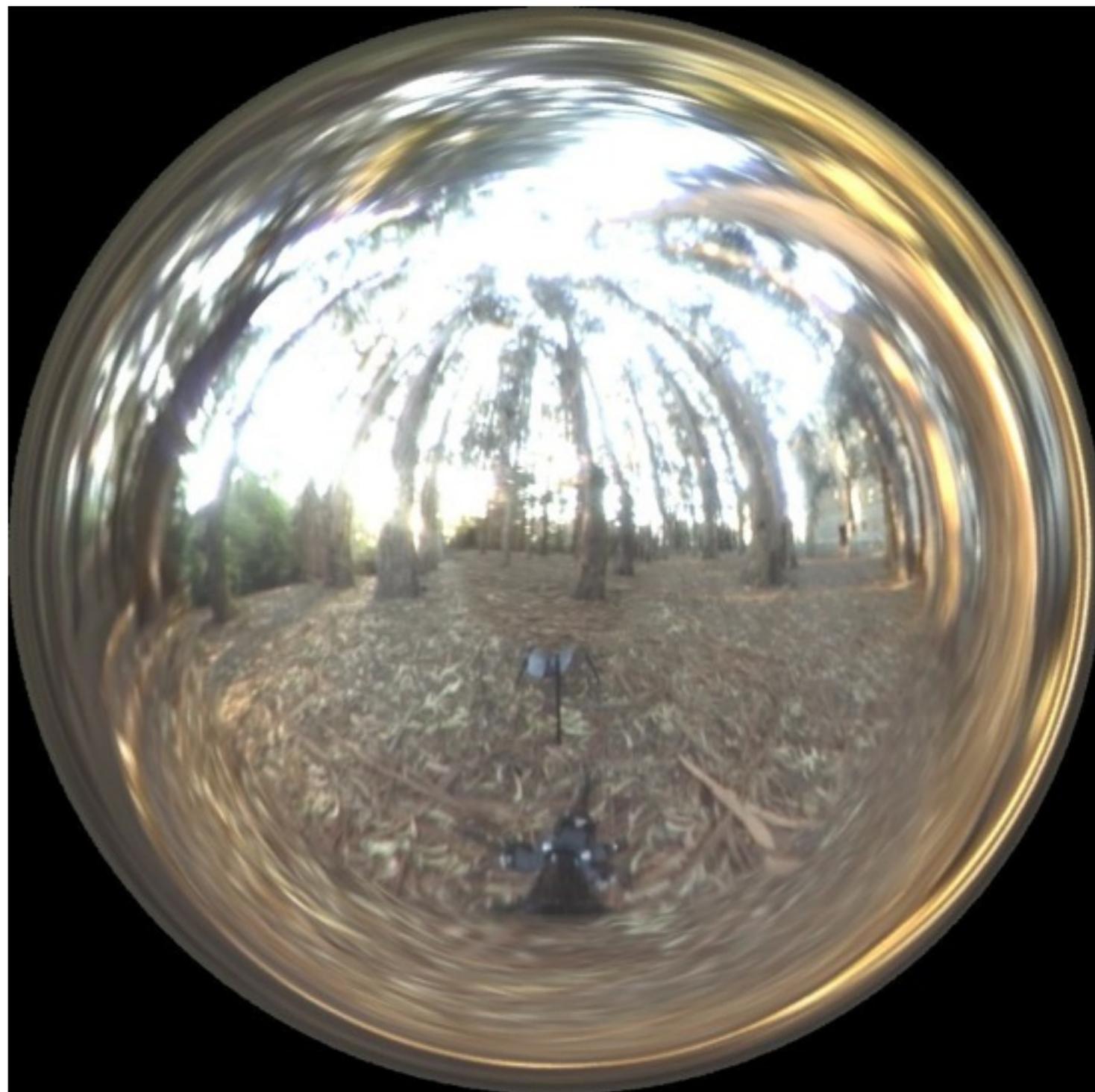


# Composer l'environnement

Avant



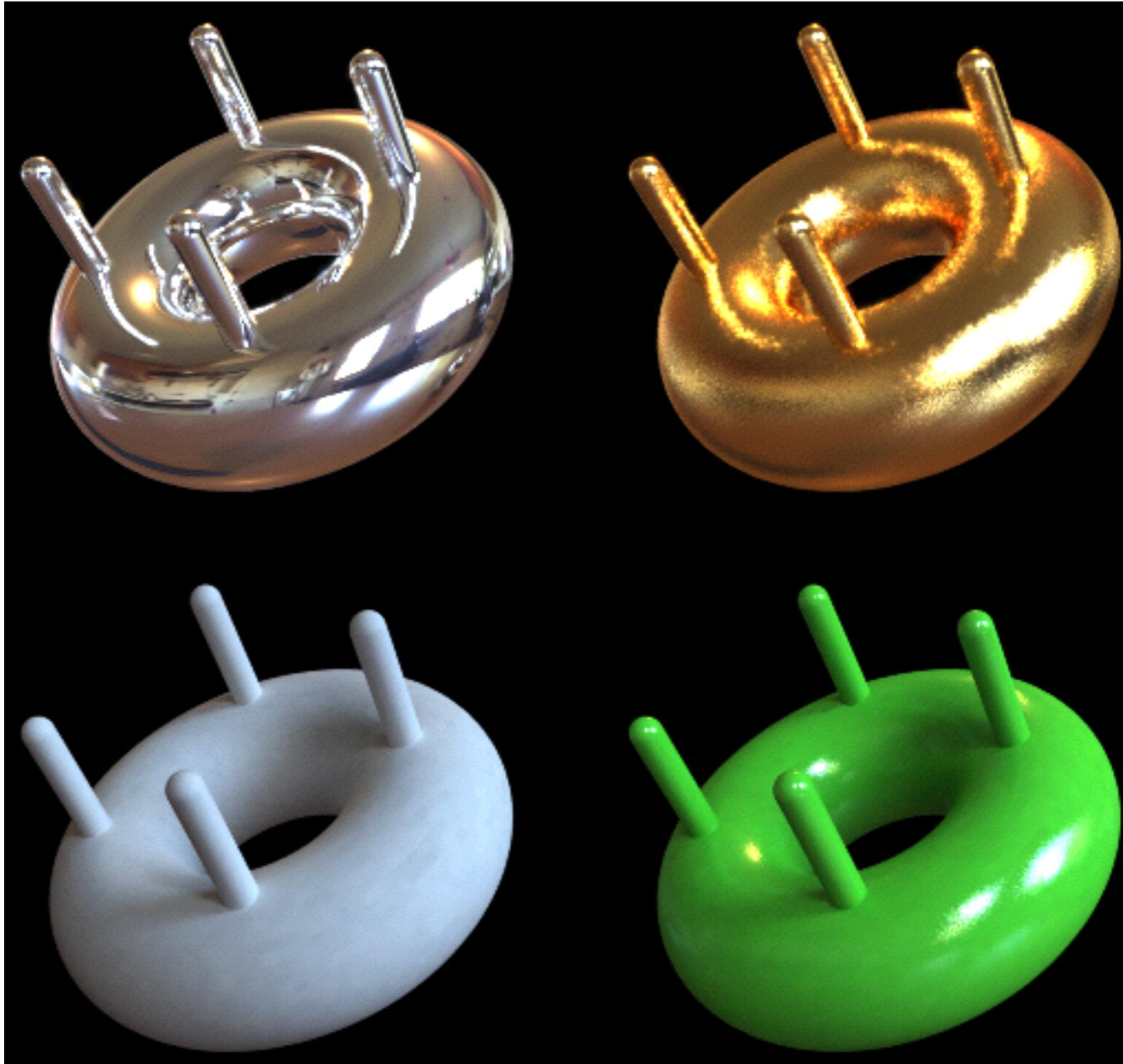
Arrière



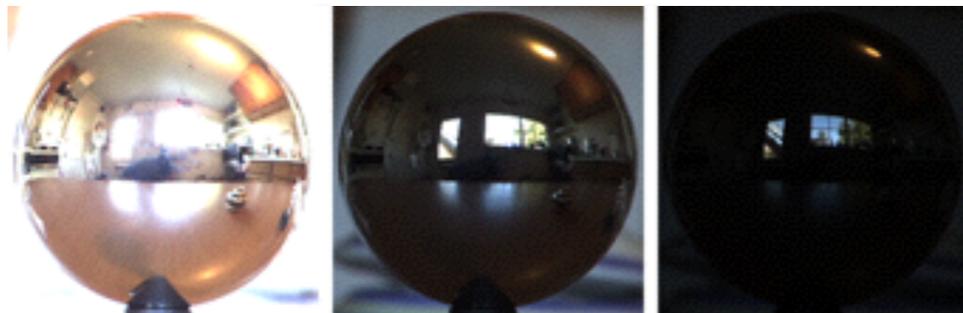
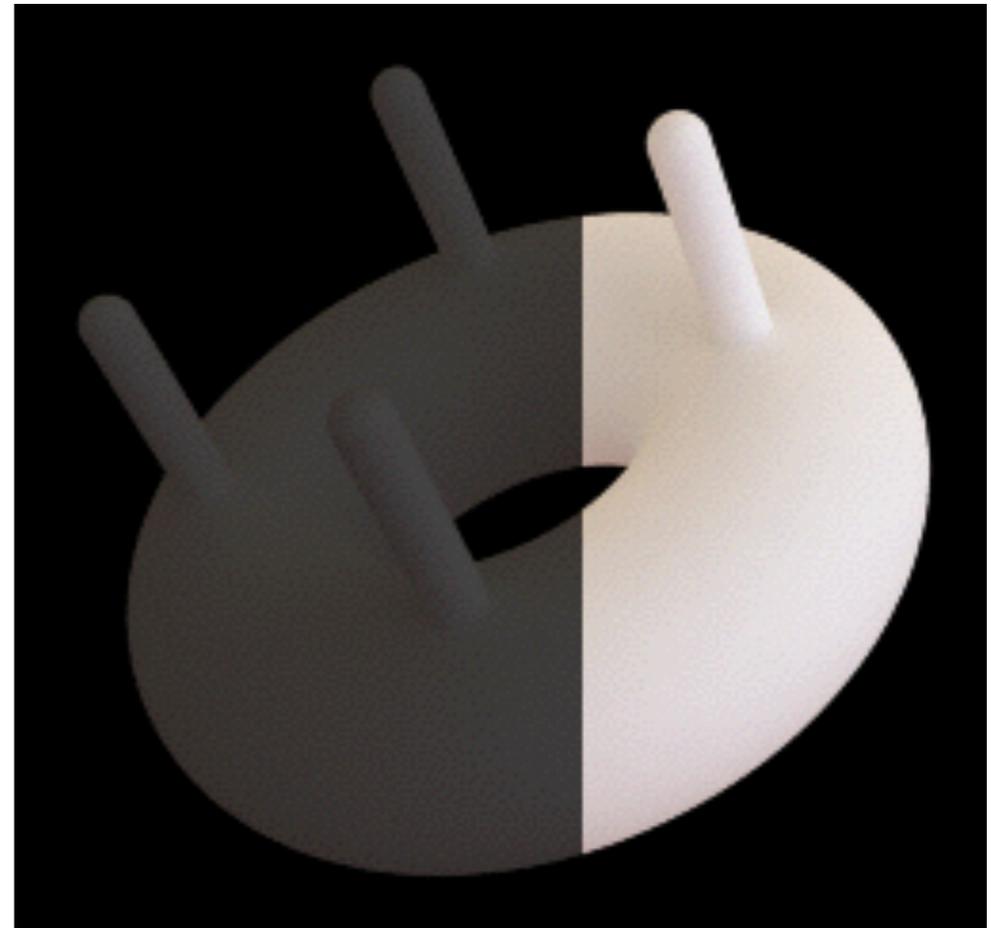
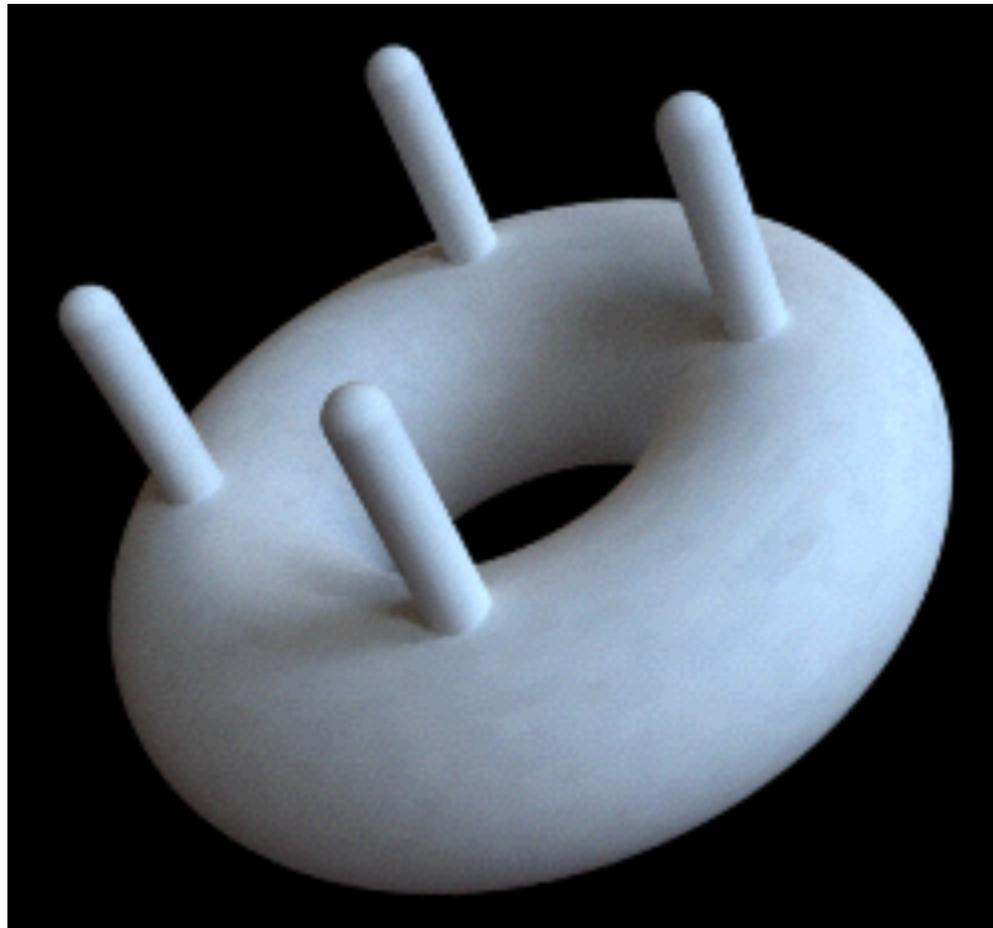
# Pas seulement pour les objets réfléchissants

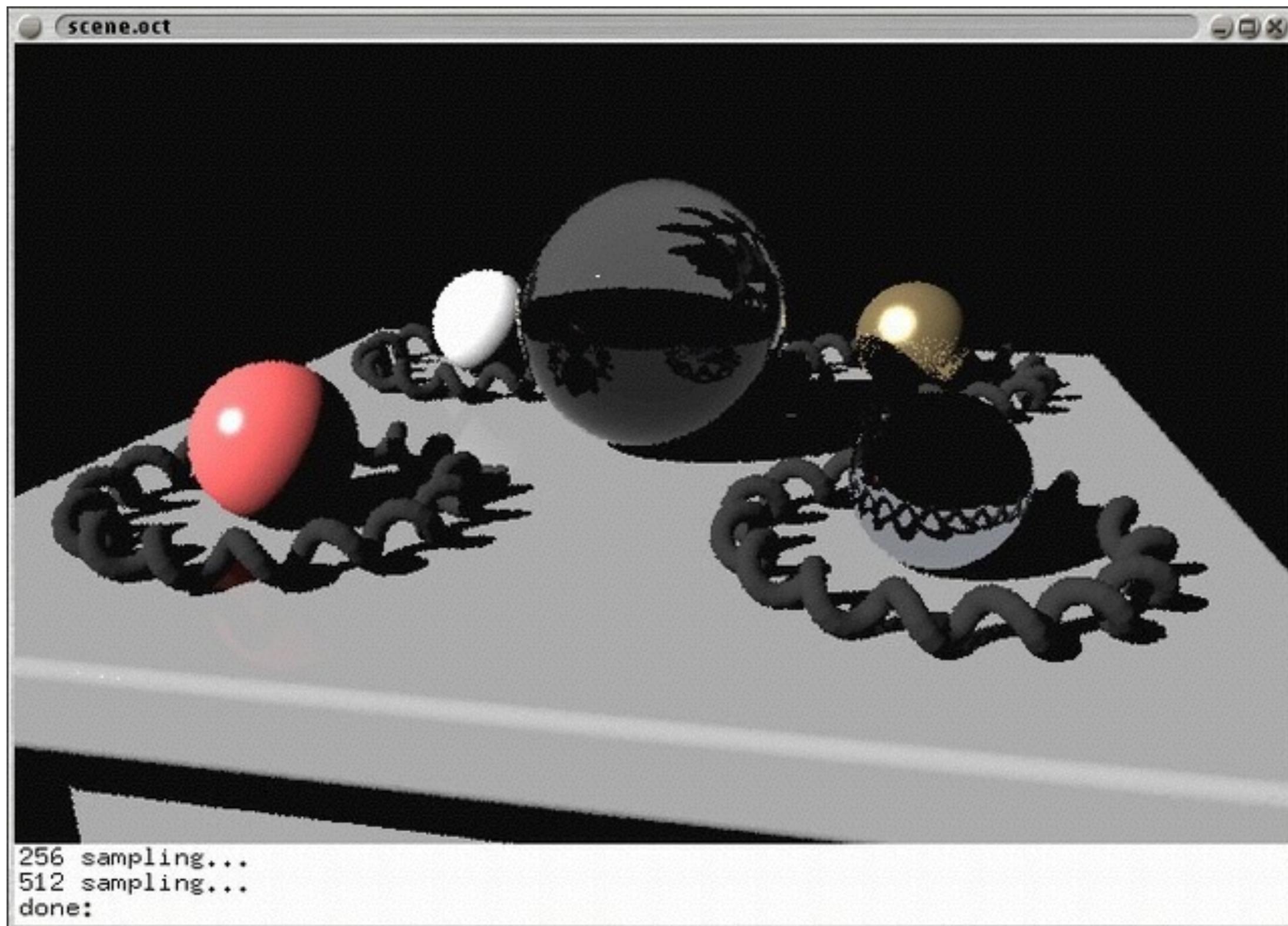
- Nous avons capté une vraie image de radiance
- Nous pouvons
  - la traiter comme une source lumineuse!
  - l'utiliser pour éclairer la scène, en simulant la propagation de la lumière dans l'environnement virtuel
  - Tous les objets peuvent être affichés (pas seulement les objets réfléchissants)!
- Quelle est le problème?

# Résultats



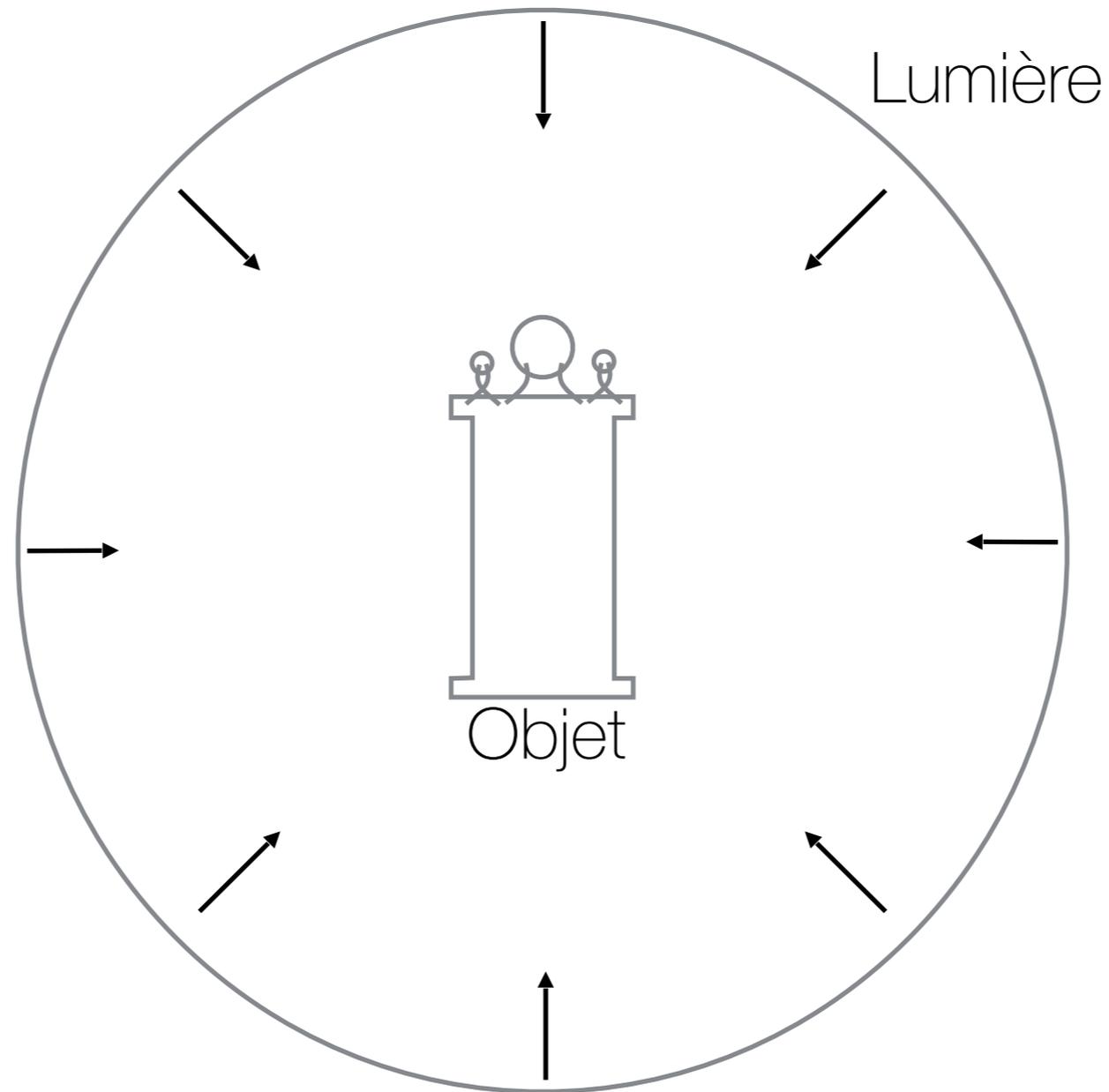
# Comparaison: radiance vs image

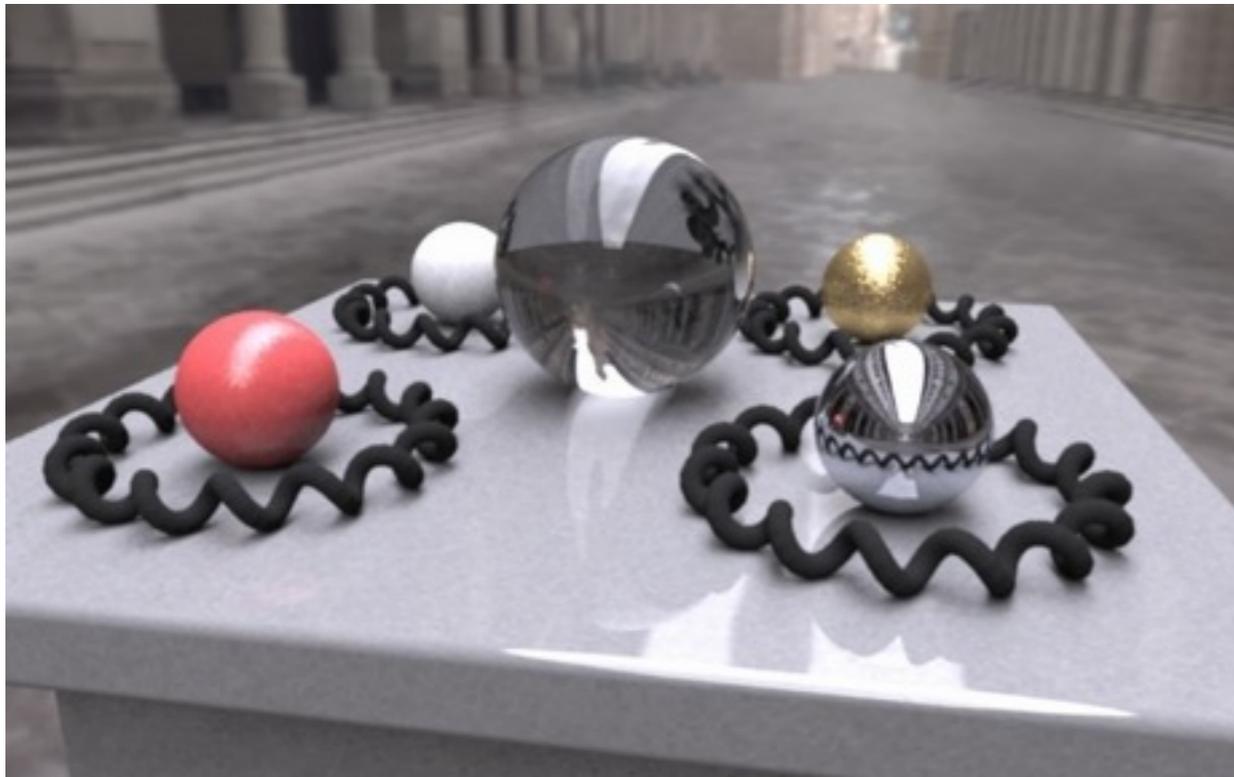
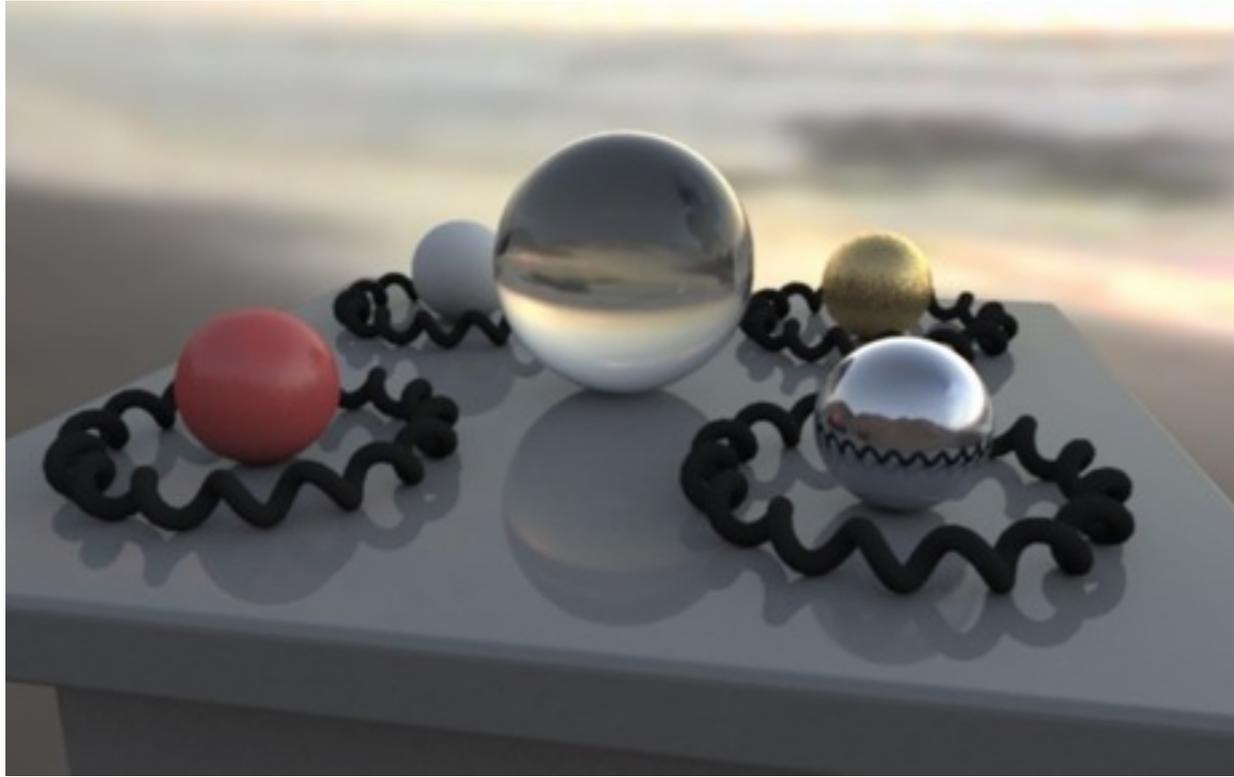


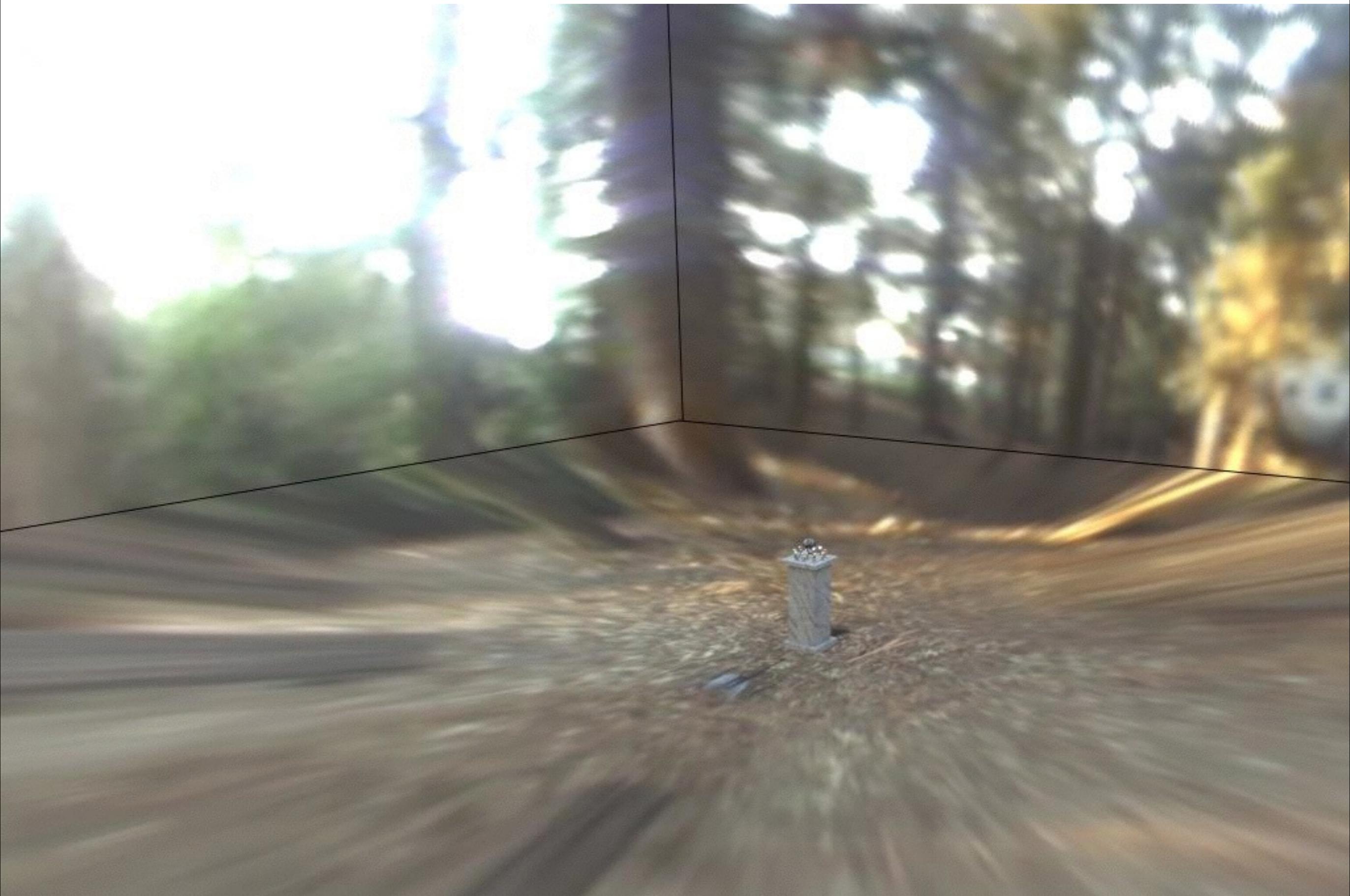


Objets virtuels illuminés par une source virtuelle

# Illuminer les objets virtuels







<http://www.pauldebevec.com/RNL/>

- Historique
  - <http://www.pauldebevec.com/ReflectionMapping/>
- Mercredi:
  - Combiner les objets virtuels avec les images réelles

# Plan — reste de la session

- Cette semaine: insertion d'objets virtuels
- Autres sujets: à votre choix!
  - “Big Data”: comprendre et synthétiser les images à partir de grandes quantités de données visuelles
  - Comment prendre de bonnes photos?
  - Comment fonctionne la kinect?
  - Caméras “lightfield”: focuser après avoir pris la photo!
    - [lytro.com](http://lytro.com)
  - Détecter les fausses images
  - ACP et visages
  - Caméras algorithmiques
  - Autres idées?



**Allez voter sur le groupe  
Facebook!**