

# M2-Images

Lumière, matières et couleurs

J.C. Iehl

October 11, 2010

## Résumé des épisodes précédents

dessiner ...

- ▶ transformations, projection,
- ▶ dessiner une primitive plane, une primitive non plane,
- ▶ visibilité : dessiner plusieurs primitives, plusieurs objets,
- ▶ + déterminer la couleur de chaque pixel.

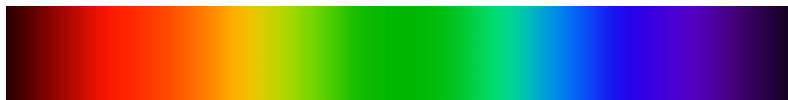
# Objectif : couleur, lumière et matières

déterminer la couleur de chaque pixel de l'image :

- ▶ la couleur d'un pixel doit représenter l'aspect de l'objet visible à travers le pixel ?
- ▶ quelques notions pour décrire l'interaction de la lumière et d'une matière,
- ▶ et déterminer une couleur.

# Domaine visible et couleurs

la couleur est la perception d'une "*énergie visible*" :

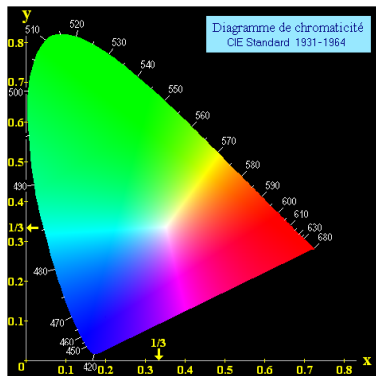


- ▶ *énergie* ? associée à une onde électromagnétique,
- ▶ *énergie visible* ? associée à l'ensemble d'ondes perçues par le système visuel,
- ▶ *domaine visible* ? ensemble d'ondes perçues.

# Domaine visible et couleurs

onde électromagnétique :

- ▶ identifiée par une *longueur d'onde*, ou une *fréquence*,
- ▶ correspond à une *teinte* ou une "couleur pure".

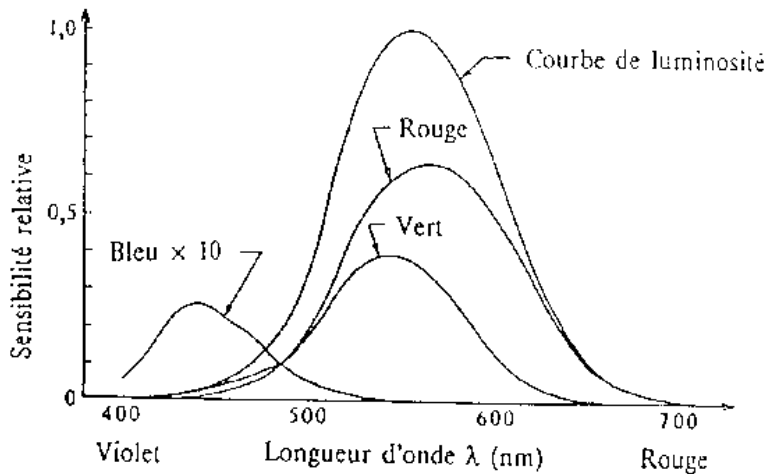


# Energie réfléchie et couleur

la couleur d'un objet :

- ▶ est la perception des énergies réfléchies par l'objet vers l'observateur,
- ▶ déterminer chaque énergie réfléchie par l'objet,
- ▶ déterminer la couleur associée à ce mélange :
- ▶ déterminer la couleur "pure" (la *teinte*) associée à chaque énergie,
- ▶ déterminer la couleur du mélange.

## Sensibilité spectrale



# Energie réfléchie et couleur

en pratique :

- ▶ on représente la distribution d'énergie sur le domaine visible par 3 valeurs moyennes,
- ▶ correspondant à notre perception : rouge, vert, bleu.

mais c'est une approximation assez grossière ...



# Objectif : un peu de lumière

selon son orientation, une surface reçoit plus ou moins de lumière :

$$L_i(p) = L_e(-\vec{l}) \cos \theta$$

- ▶  $L_e(-\vec{l})$  : énergie émise par la source de lumière vers  $p$ , dans la direction  $-\vec{l}$ ,
- ▶  $\vec{n}_p$  : normale de la surface au point  $p$ ,
- ▶  $\cos \theta = \max(0, \vec{l} \cdot \vec{n}_p)$ ,  $\theta$  est angle entre  $\vec{l}$  et  $\vec{n}_p$ .

selon sa matière et son orientation, une surface réfléchit plus ou moins de lumière vers l'observateur : notation  $f_r(\vec{l}, p, \vec{o})$ .

# Objectif: un peu de lumière

schema + notations.

# Objectif : un peu de lumière

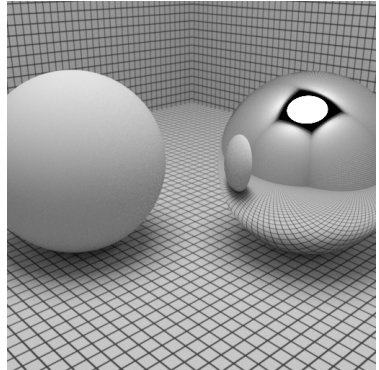
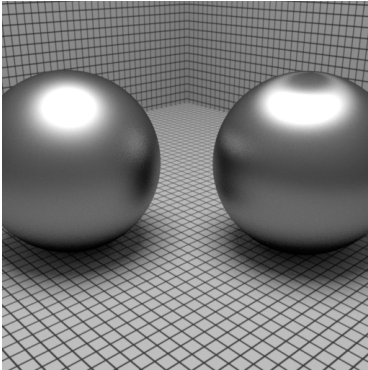
description de la matière (BRDF) :

- ▶ aspect mat / diffus :  $f_r(\vec{l}, \rho, \vec{o}) = \frac{0 < \text{constante} < 1}{\pi}$ ,
- ▶ aspect réfléchissant:  $f_r(\vec{l}, \rho, \vec{o}) = \frac{m+2}{2\pi} \cos^m \theta_h$ ,
- ▶ avec  $\vec{h} = \frac{1}{2}(\vec{l} + \vec{o})$  et  $\theta_h$  l'angle entre  $\vec{n}_p$  et  $\vec{h}$ ,
- ▶  $\cos \theta_h = \max(0, \vec{h} \cdot \vec{n}_p)$

énergie réfléchie vers l'observateur :

- ▶  $L_r(\rho, \vec{o}) = L_e(-\vec{l}) f_r(\vec{l}, \rho, \vec{o}) \cos \theta$

# Exemples :



# Objectif : un peu de lumière

schema + notations.

quelle couleur pour une matière  $f_r(\vec{l}, p, \vec{o})$  et une source de lumière  $L_e(\vec{v})$  ?

## En pratique :

- ▶ une source ponctuelle :  $q$ ,
- ▶ sa quantité d'énergie émise  $L_e$ , un triplet  $(r, g, b)$ ,
- ▶ un point  $p$ , sa normale  $\vec{n}_p$ ,
- ▶ et sa matière :  $f_r(\vec{l}, p, \vec{o})$ .

calculer la couleur du pixel sur lequel se projette  $p$ .

## En pratique :

comment décrire la "couleur" d'une matière ?

encore une approximation :

on utilise la couleur de l'énergie réfléchiée par la matière lorsqu'elle est éclairée par une source "blanche" !

la matière est décrite par un triplet :  $(r, g, b) \cdot f_r(\vec{l}, p, \vec{o})$

## Objectif : un peu d'ombre

comment déterminer qu'un point est éclairé par une source de lumière ?

idée :

"tester" la visibilité du point et (d'un point à la surface) de la source.

- ▶ si un autre objet masque la source, le point est à l'ombre,
- ▶ sinon, le point est éclairé, il reçoit la lumière émise par la source.

s'il y a un autre objet sur la droite entre le point et la source, le point ne reçoit pas de lumière.



## Objectif : un peu de reflet

la lumière se réfléchit sur un miroir uniquement dans la direction  
 $\vec{r} = 2(\vec{n} \cdot \vec{l}) \vec{n} - \vec{l}$ .

calculer l'énergie réfléchiée par le miroir vers l'observateur ?

idée :

elle dépend de l'objet visible dans la direction  $\vec{r}$ .

trouver le point visible dans la direction  $\vec{r}$  et déterminer l'énergie réfléchiée.

## Objectif : un peu de reflet

et s'il y a plusieurs miroirs ?

on recommence ...

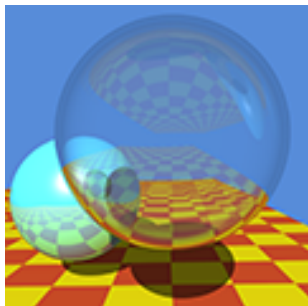
l'algorithme devient récursif :

- ▶ pour connaître l'énergie arrivant depuis le point visible dans la direction  $\vec{r}$ ,
- ▶ il faut connaître l'énergie arrivant sur ce point, qui peut lui même être sur un miroir ... qui peut ...

# Objectif : un peu de transparence

et si l'objet est transparent ?

- ▶ l'énergie réfléchiée par un point à la surface d'un objet transparent dépend de l'énergie dans la direction  $\vec{r}$ ,
- ▶ et de l'énergie dans la direction  $\vec{t}$



# Objectif : ombres

rappel :

un point ne reçoit de la lumière que si la source de lumière est directement visible.

détails :

- ▶ "Rendering antialiased shadows with depth maps"  
B. Reeves, D. Salesin, R.L. Cook, 1987
- ▶ "Casting curved shadows on curved surfaces"  
L. Williams, 1978

# Objectif : ombres

sachant que :

- ▶ un point est éclairé si la source de lumière est visible,
- ▶ == un point est éclairé s'il est le plus proche de la source dans la direction  $\vec{l}$ ,

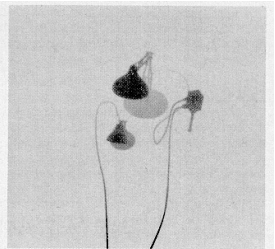
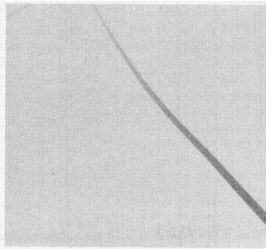
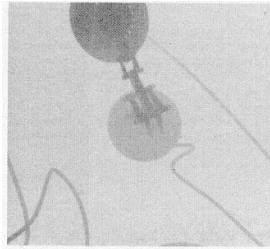
comment trouver la distance des autres objets pour la direction  $\vec{l}$ ?

# Objectif : ombres

dessiner les objets :

- ▶ du point de vue de la source ...
- ▶ puis, comparer la profondeur de chaque pixel sur la direction  $\vec{l}$ , pendant le dessin depuis l'observateur.

# Exemple : dessiner depuis la source



# Objectif : ombres

en 2 étapes :

- ▶ 1. dessiner depuis la source, conserver le z-buffer,
- ▶ 2. dessiner depuis l'observateur, transformer chaque point dans le repère *Fenêtre* attaché à la source,
- ▶ comparer la profondeur du point avec le z-buffer de la source,
- ▶ si la profondeur est plus petite que la valeur dans le z-buffer de la source :
- ▶ le point est visible de la source, il reçoit la lumière émise par la source,
- ▶ sinon, il est à l'ombre de la source.



# Objectif : ombres

quelques détails à régler :

- ▶ identifier l'ensemble d'objets projetant une ombre visible,
- ▶ déterminer les paramètres pour dessiner du point de vue de source,
- ▶ choisir une résolution pour le z-buffer ?
- ▶ filtrer les résultats des tests sur la profondeur ?