

TEXTURES

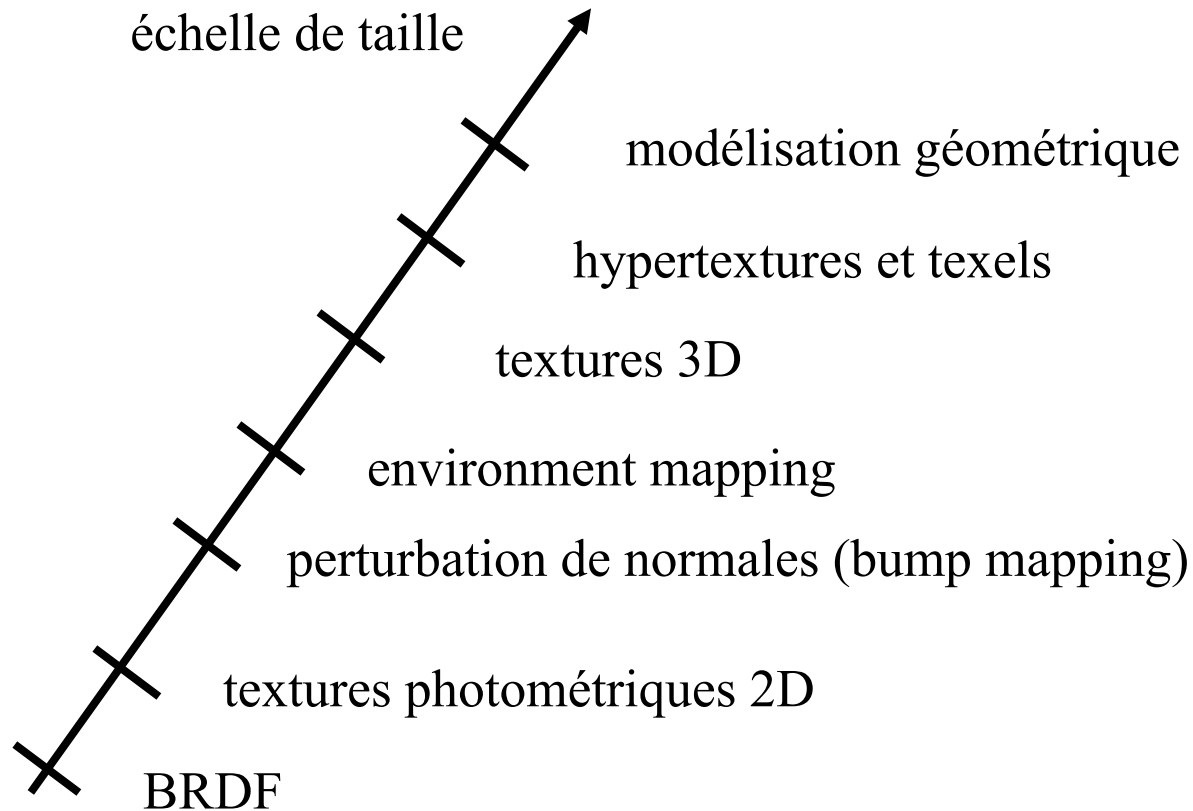
1 - Introduction

texture = fonction (de \mathbb{R}^2 ou de \mathbb{R}^3) permettant de moduler l'aspect **photométrique** ou **géométrique** en tout point de la surface d'un objet donné

deux problèmes :

- la génération
- la transformation texture-objet

La notion de texture est liée à la notion d'échelle :

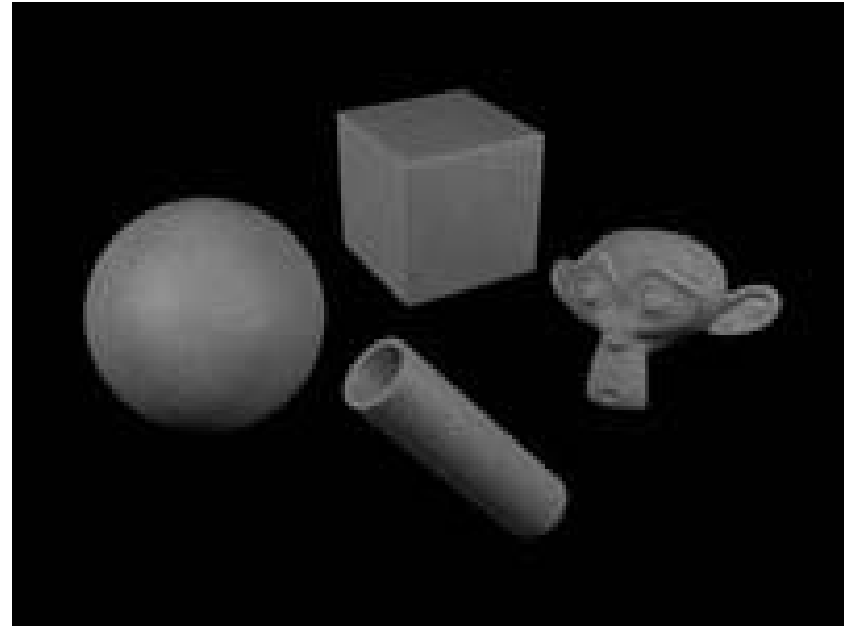


1.1 - Textures 2D

coller des images $I(u,v)$ sur la surface S des objets

\Rightarrow associer à chaque point $P(x,y,z)$ de S un pixel de $I(u,v)$





1.2 – Placage d’environnement (environment mapping) : pour simuler les effets de réflexion calculer le vecteur de réflexion $R(x,y,z)$ (par rapport à la normale et au vecteur de visée)

exprimer R en coordonnées sphériques :

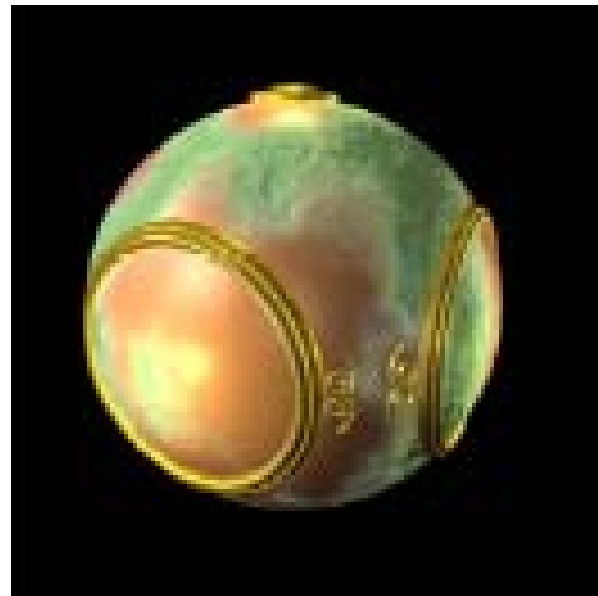
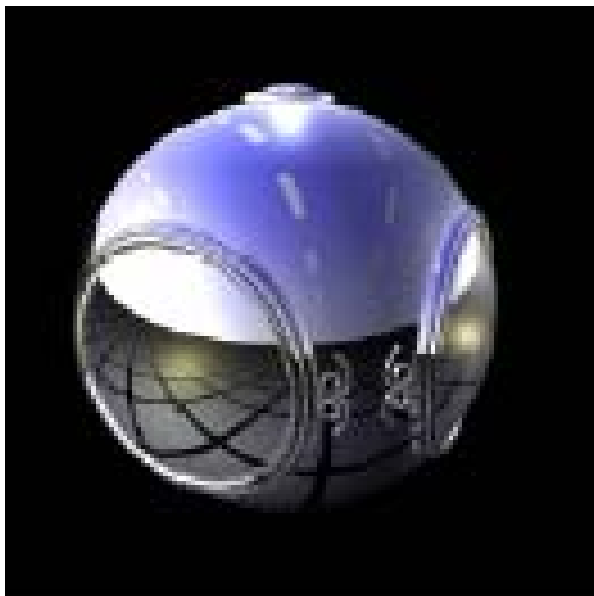
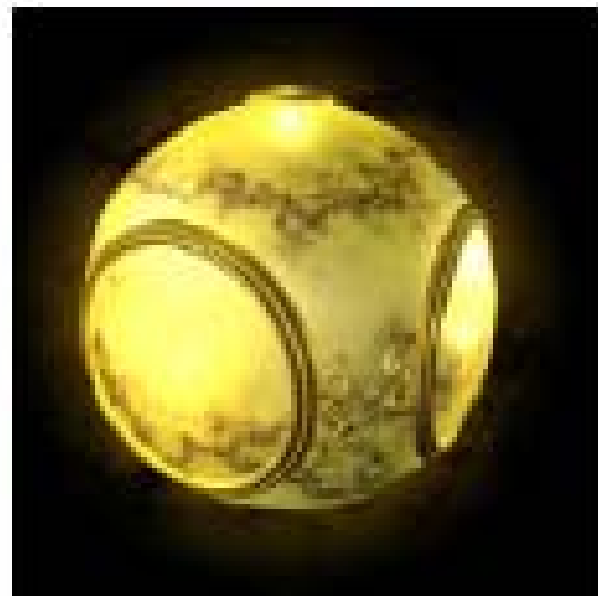
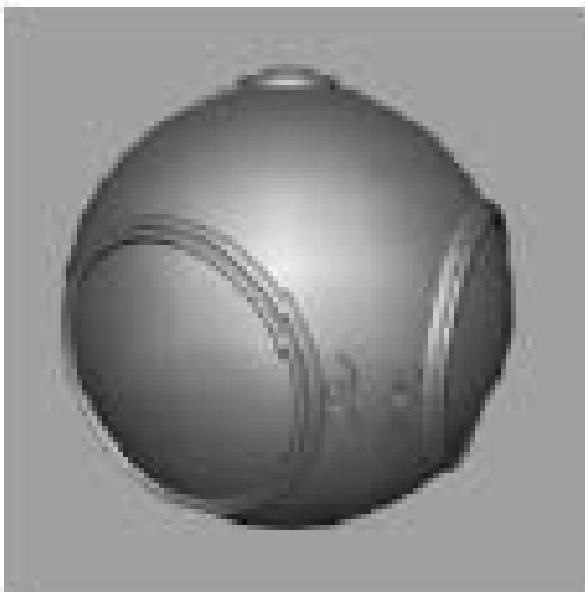
$$\phi = \sin^{-1}(z)$$

$$\theta = \tan^{-1}(y/x)$$

utiliser ces angles pour accéder aux pixels de l’image $I(u,v)$



(Blinn)





(Heidrich and Seidel)

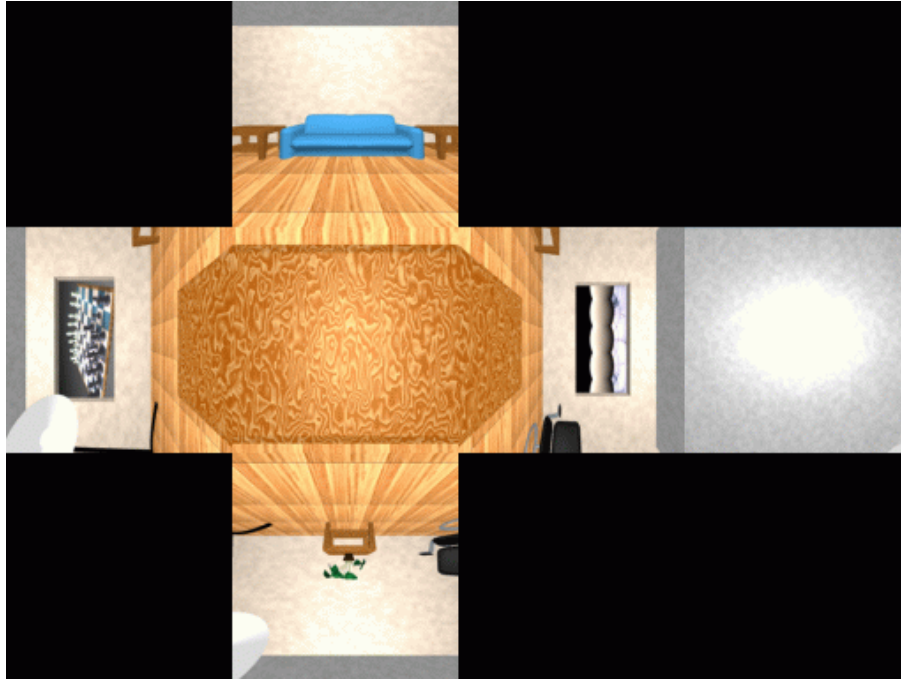


environment mapping



lancer de rayons

Exemple de “Cubic Map” “



(S. Cheney)

1.3 - Perturbation de normales : pour simuler des surfaces rugueuses ou bosselées

utiliser une texture $h(u,v)$ de variation des hauteurs au lieu de variation des couleurs

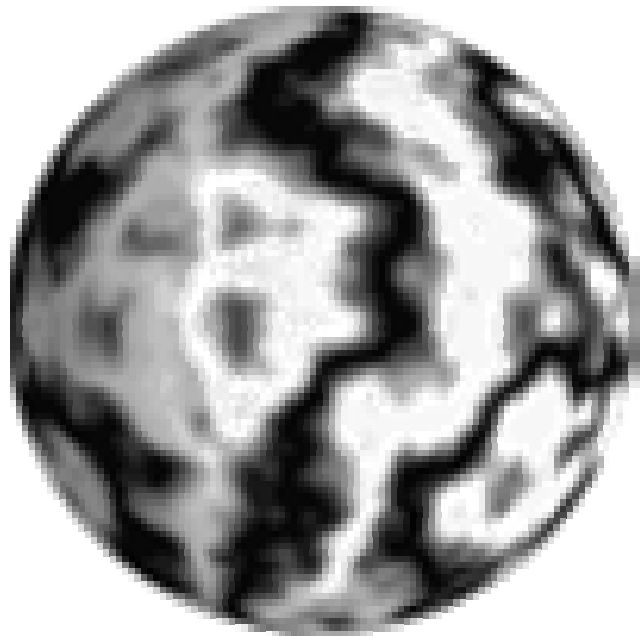
=> déplacer chaque point le long de sa normale

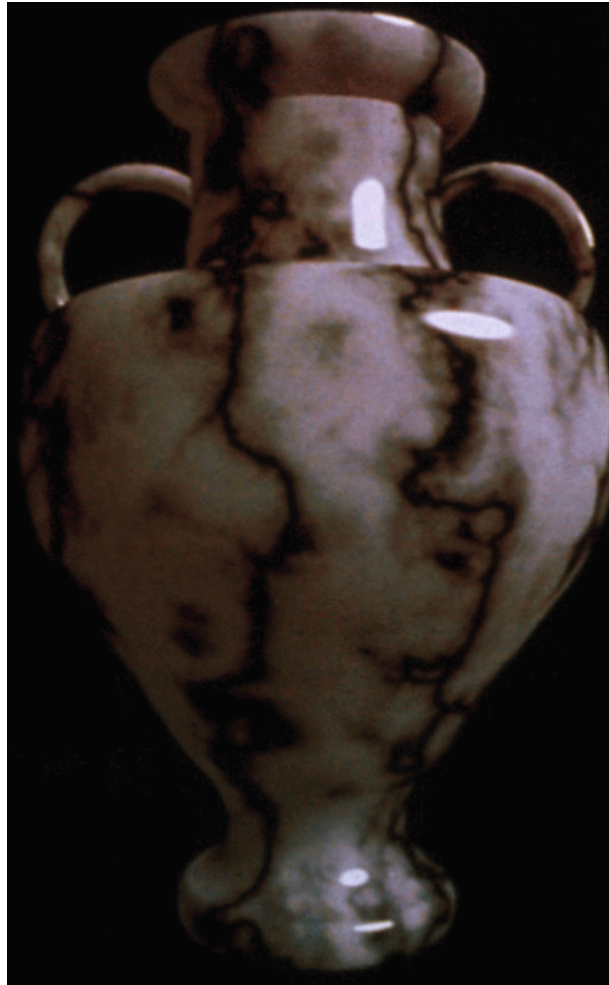
\vec{N} normale à la surface

$$\vec{N}' = \vec{N} + h(u, v) \frac{\vec{N}}{\|\vec{N}\|}$$

1.4 - Textures 3D

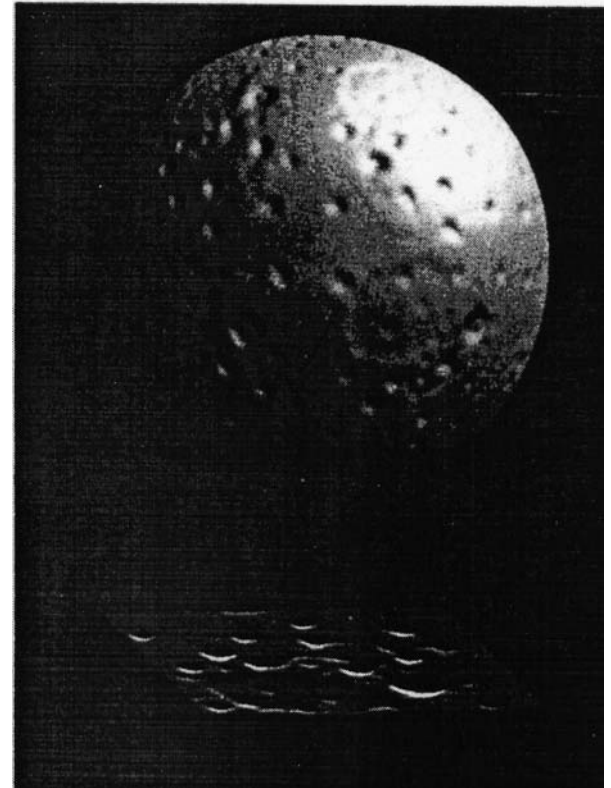
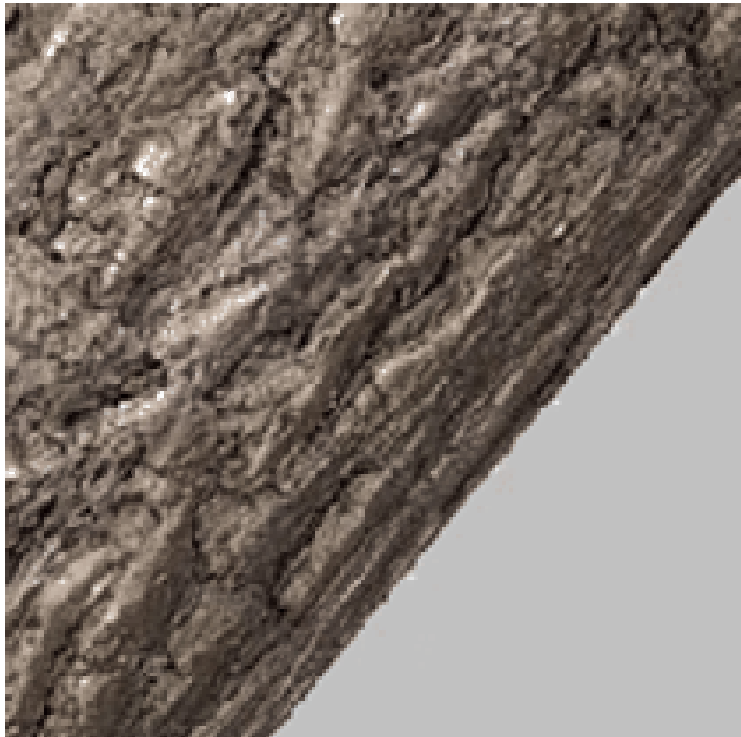
consiste à tailler les objets dans un bloc 3D de matière
=> à chaque point de l'espace 3D est associé une couleur





(K. Perlin)

remarque : on peut aussi perturber les normales en utilisant un champ de vecteurs (on ajoute un vecteur à la normale)



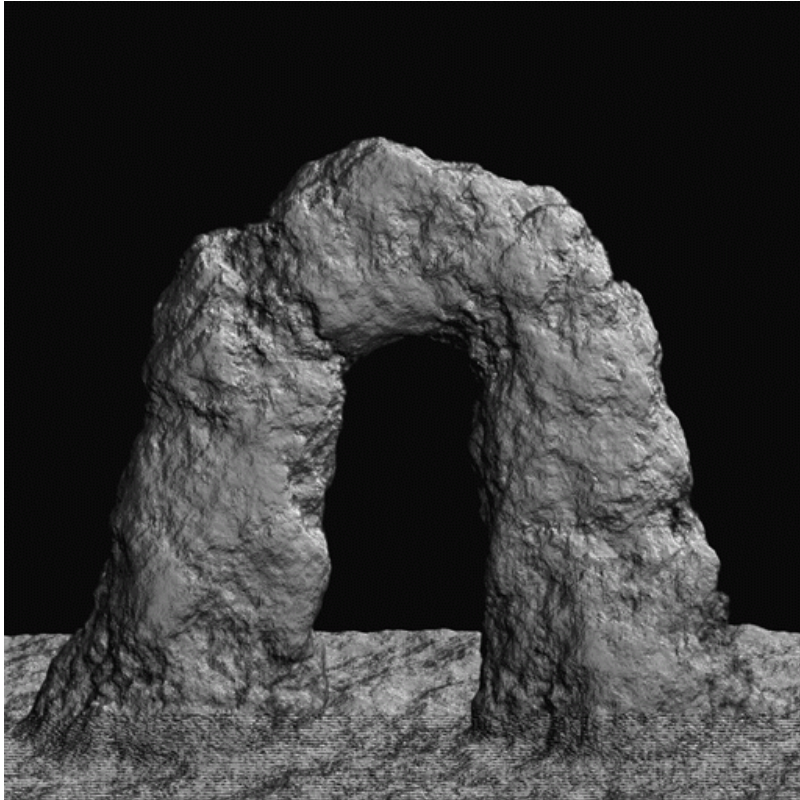


1.5- Hypertextures

on modélise les objets en faisant varier la densité de matière

$$D : (x, y, z) \rightarrow [0,1]$$

on module cette densité par une fonction aléatoire (bruit, turbulence, ...) là où la matière n'est pas opaque



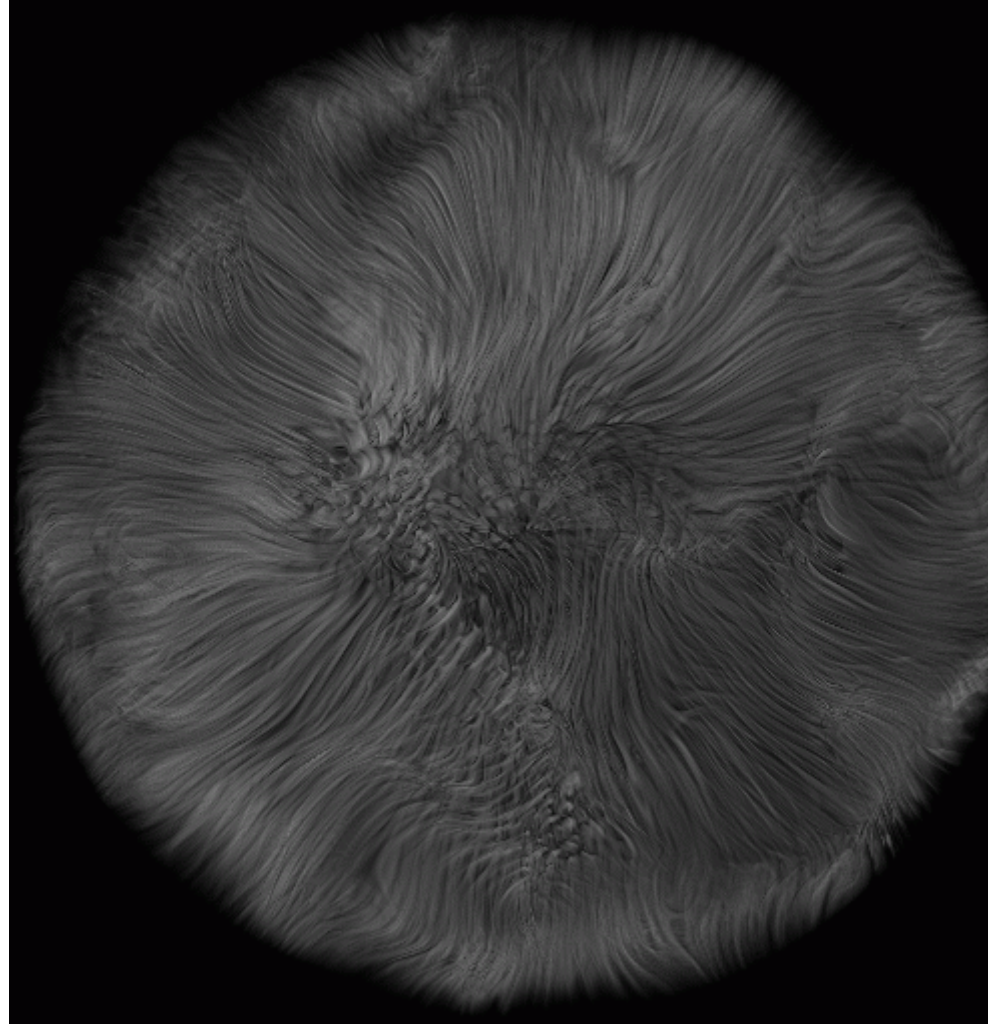
1.6- Texels

$$\text{texel} = \begin{cases} \text{une fonction de densité } d(x, y, z) \\ \mathbf{B} = (\vec{n}(x, y, z), \vec{t}(x, y, z), \vec{b}(x, y, z)) \text{ en tout point du texel} \\ \text{une fonction de réflexion} \end{cases}$$

on plaque des parallélépipèdes rectangle
(texels) sur les objets comme pour le placage
2D (on rajoute une épaisseur aux objets)

(Kajiya)





comment faire le choix ?

❑ textures 2D pour les surfaces déformables, le papier peint, les vêtements ;

❑ texture 3D pour les objets rigides dont la nature même est 3D (marbre, bois, ...) ;

❑ texels : comme les textures 2D, mais avec une couche micro-géométrique au lieu d'une couleur (fourrure, champs, ...) ;

❑ hypertextures : on considère l'ensemble de l'objet (roche, coton, flammes, ...)

2- Génération

❑ numérisation

problèmes :

éclairage non uniforme

taille limitée (visibilité des carreaux)

continuité non assurée

difficulté en 3D

❑ méthodes procédurales

couleur = **si** ($\text{frac}(x) < 0.5$ **et** $\text{frac}(y) < 0.5$) **ou** ($\text{frac}(x) > 0.5$ **et** $\text{frac}(y) > 0.5$)
alors couleur_1
sinon couleur_2

➤ méthode spectrale

$$T(Y, Y, Z) = \sum_{i=1}^n C_i \frac{\sin(\omega_i x + P x_i) + 1}{2} \times \sum_{i=1}^n C_i \frac{\sin(\omega_i y + P y_i) + 1}{2}$$

➤ bruit

$$\text{bruit}(x, y, z) = \sum_{i=|x|}^{|x|+1} \sum_{j=|y|}^{|y|+1} \sum_{k=|z|}^{|z|+1} N_{i,j,k} (x-i, y-j, z-k)$$

$$N_{i,j,k} (u, v, w) = f(u) f(v) f(w) (G_{i,j,k} (u, v, w) + V_{i,j,k})$$

$$f(t) = \text{si } \text{abs}(t) < 1 \quad \text{alors } 2\text{abs}(t^3 - 3t^2 + 1) \quad \text{sinon } 0$$

$G_{i,j,k}$ et $V_{i,j,k}$ vecteur et scalaire pseudo - aléatoires

➤ turbulence

$$\text{turb}(x) = \sum_{i=1}^n \text{abs}(\text{bruit}(2^i X)) / 2^i$$

exemples :

$$\textit{marbre}(x, y, z) = \sin(x + \textit{turb}(x, y, z))$$

$$\textit{bois}(x, y, z) = \sin\left(\sqrt{x^2 + y^2} + \textit{bruit}(x, y, z)\right)$$

avantages :

- beaucoup de types de textures
- définition continue sur tout l'espace
- définition compacte

inconvénients :

- nombreux essais
- coûteux en calculs

❑ **modèles naturels** : utilisation des lois d 'autres sciences
(physique, chimie, biologie, ...)

avantages :

résultats très réalistes

inconvénients :

il faut connaître un modèle

la simulation peut être coûteuse

❑ **méthodes par analyse** : consiste à partir d'un résultat visuel et à extraire des informations pour régénérer une texture similaire.

Il existe plusieurs types de méthodes :

➤ *statistiques* : utilisent en général la matrice de co-occurrences
$$P(i,j,d,\phi) = \#\{[(k,l),(m,n)] / k-m = d, l-n = \phi, \\ M(k,l) = i, M(m,n) = j\}$$

➤ *pyramidales* : décomposition de la texture de manière pyramidale et à chaque étage étude de l'histogramme

➤ *spectrales* : basées sur le traitement du signal (Fourier). On extrait des fréquences significatives, on fait une somme de cosinus correspondant aux fréquences sélectionnées, on module les amplitudes et les phases des cosinus

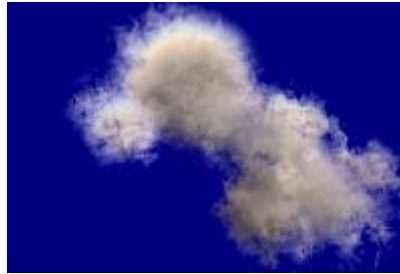
applications : mer, veines du bois, ...

☐ autres méthodes

➤ méthodes fractales

principe dans le cas mono-dimensionnel

$A_1(a_1, v_1), A_2(a_2, v_2) \Rightarrow M(, v)$ avec $v = +$ échelle $(2^{-h})^n X$



(Musgrave)

➤ **méthodes structurelles**

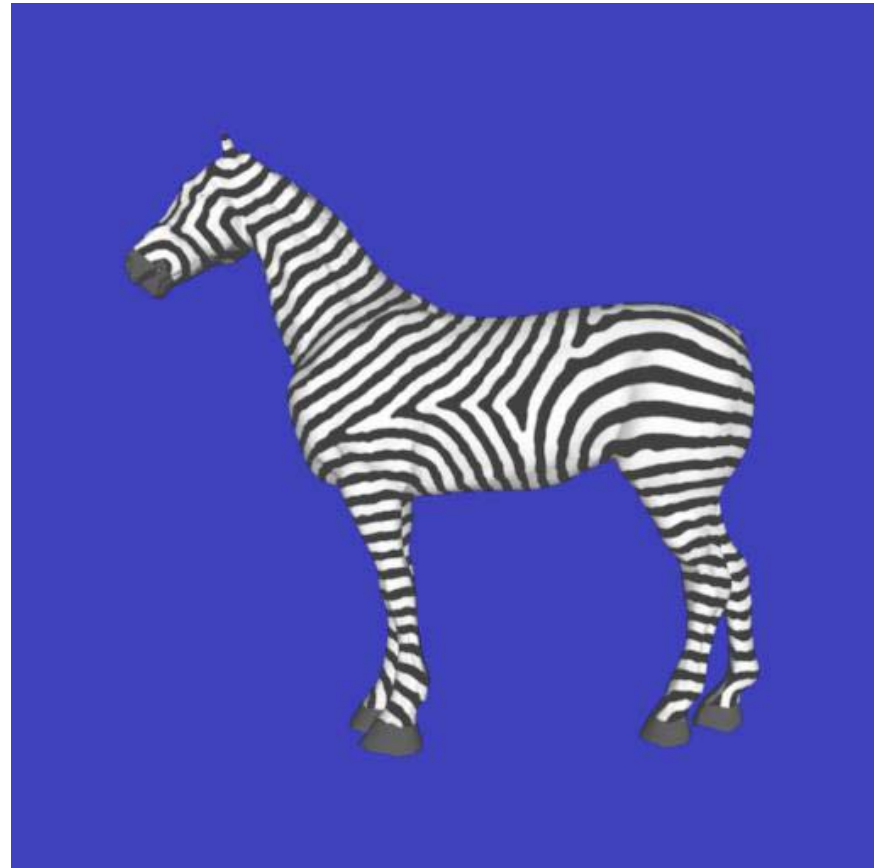
➤ **génération par évolution**

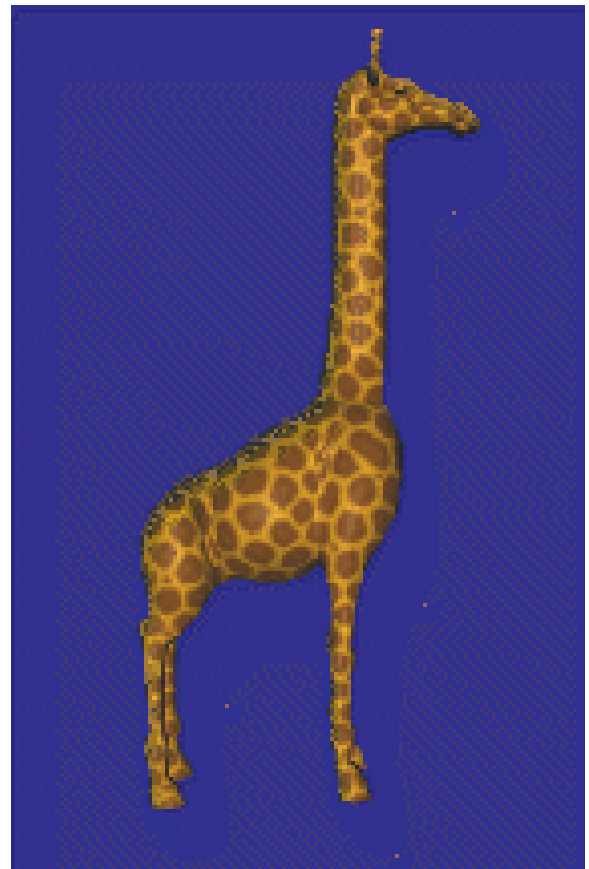
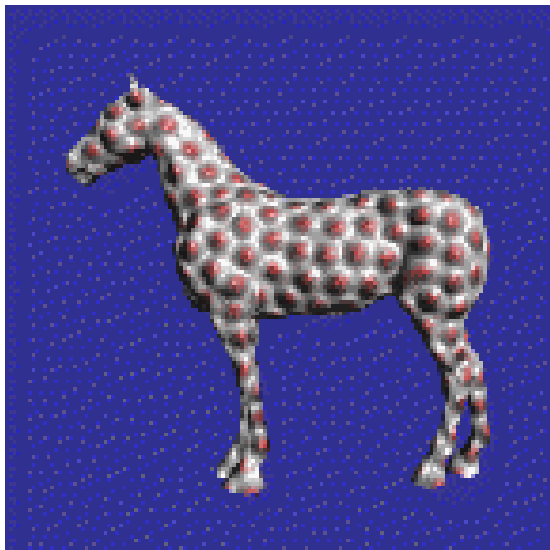
- ✓ sélection
- ✓ mutation
- ✓ reproduction

➤ **génération par réaction-diffusion**

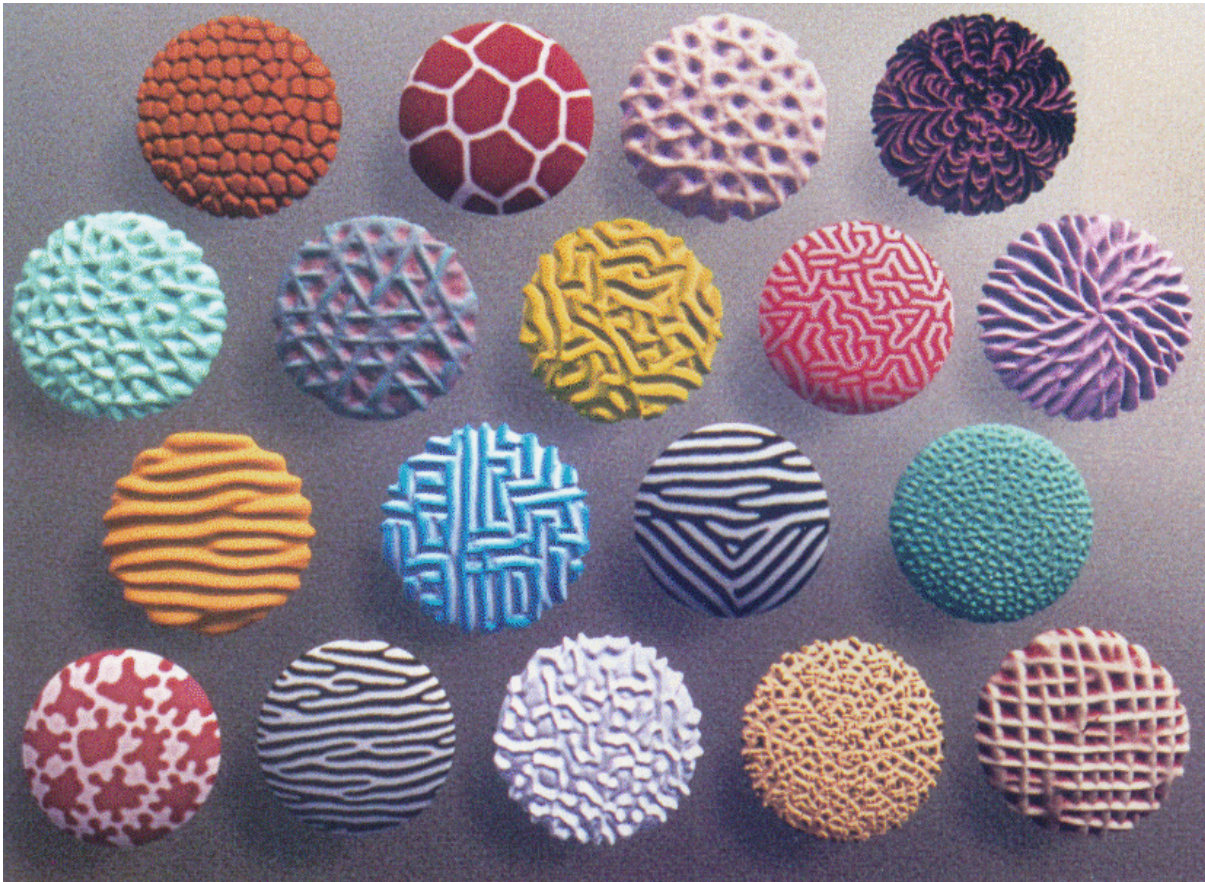
$C(x,y,z)$ concentration d'un morphogène sur une surface

$$\frac{\partial C}{\partial t} = a^2 \nabla^2 C - bC + R$$



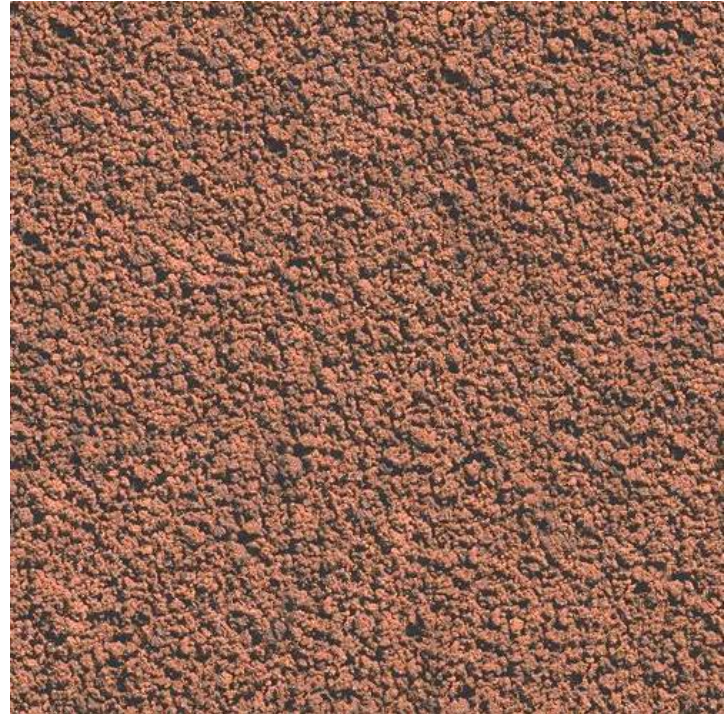


(Greg Turk)

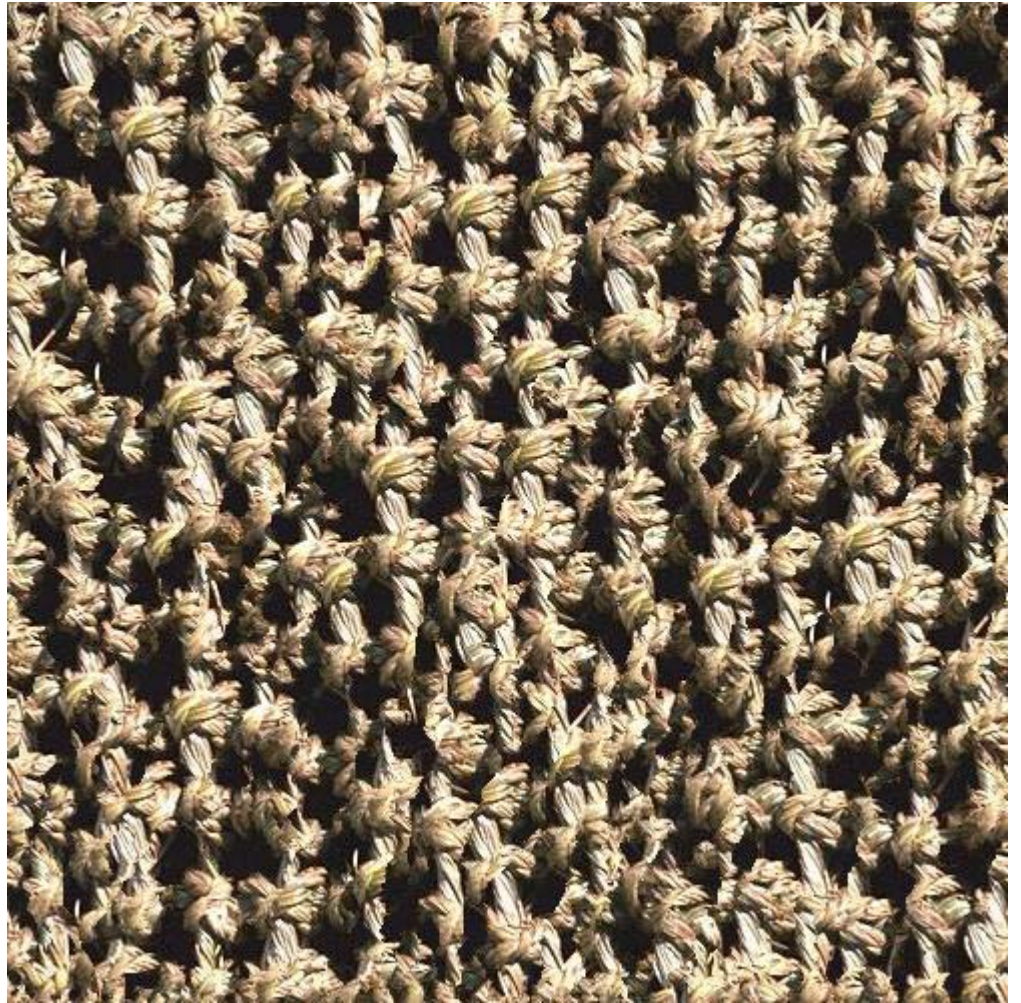


➤ **modèles particuliers : flammes,**

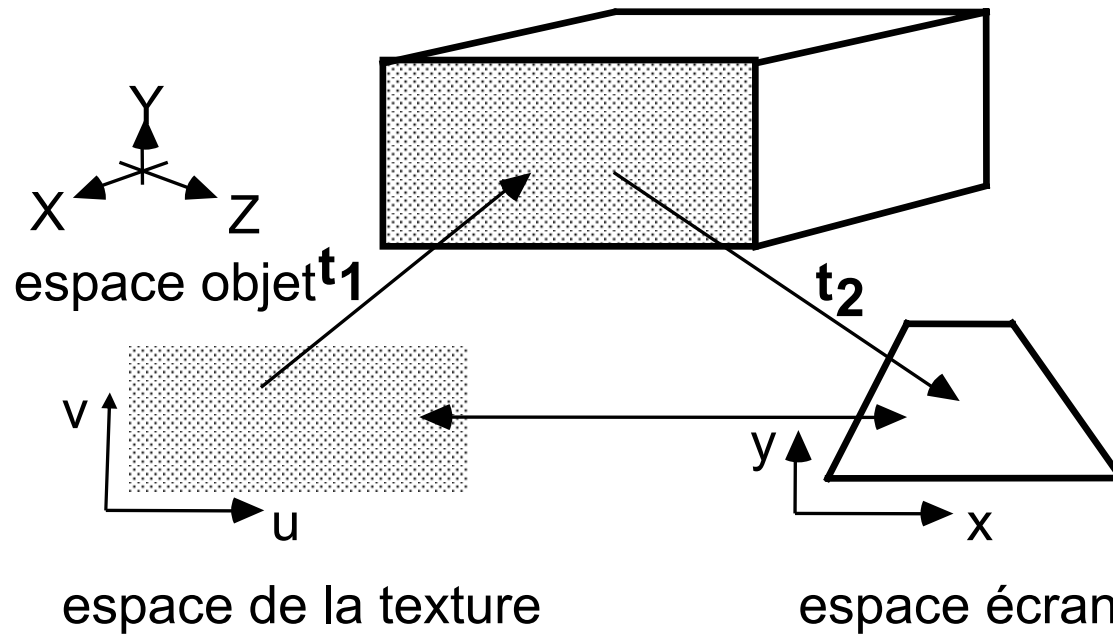
➤ **modèles par analyse**







3 - Transformation texture-objet (placage)



transformation directe :

$$F : R^2 \rightarrow R^2$$

$$(u, v) \rightarrow (x, y) = (k(u, v), l(u, v))$$

$$I_a(u, v) = T_a(u, v)$$

transformation inverse :

$$F^{-1} : R^2 \rightarrow R^2$$

$$(x, y) \rightarrow (u, v) = (f(x, y), g(x, y))$$

$$I_a(x, y) = T_a(u, v)$$

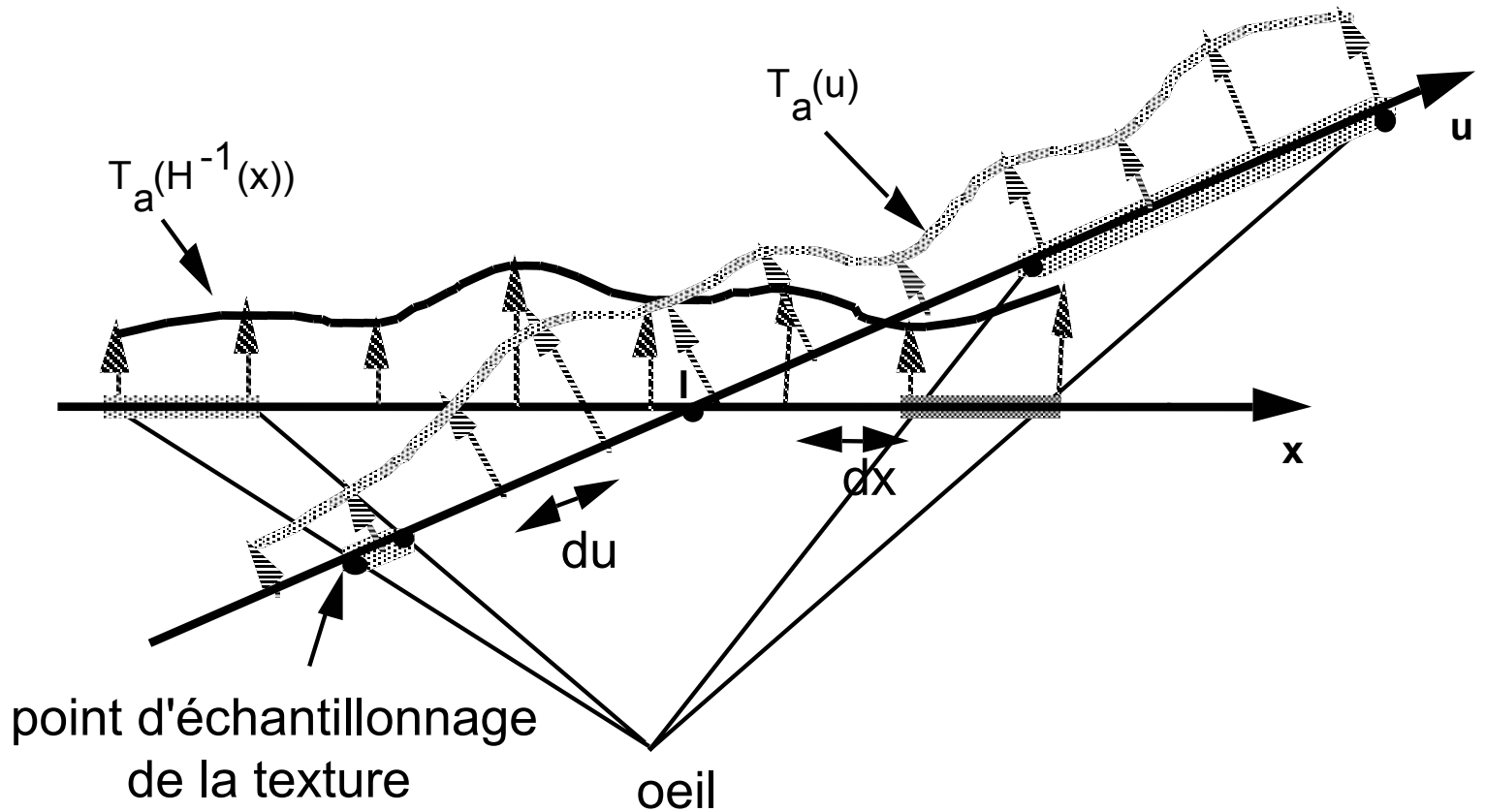
exemple : si on met en perspective une texture plane sur une surface plane, F^{-1} est une homographie :

$$u = f(x, y) = \frac{ax + by + c}{mx + ny + p}$$

$$v = g(x, y) = \frac{kx + hy + l}{mx + ny + p}$$

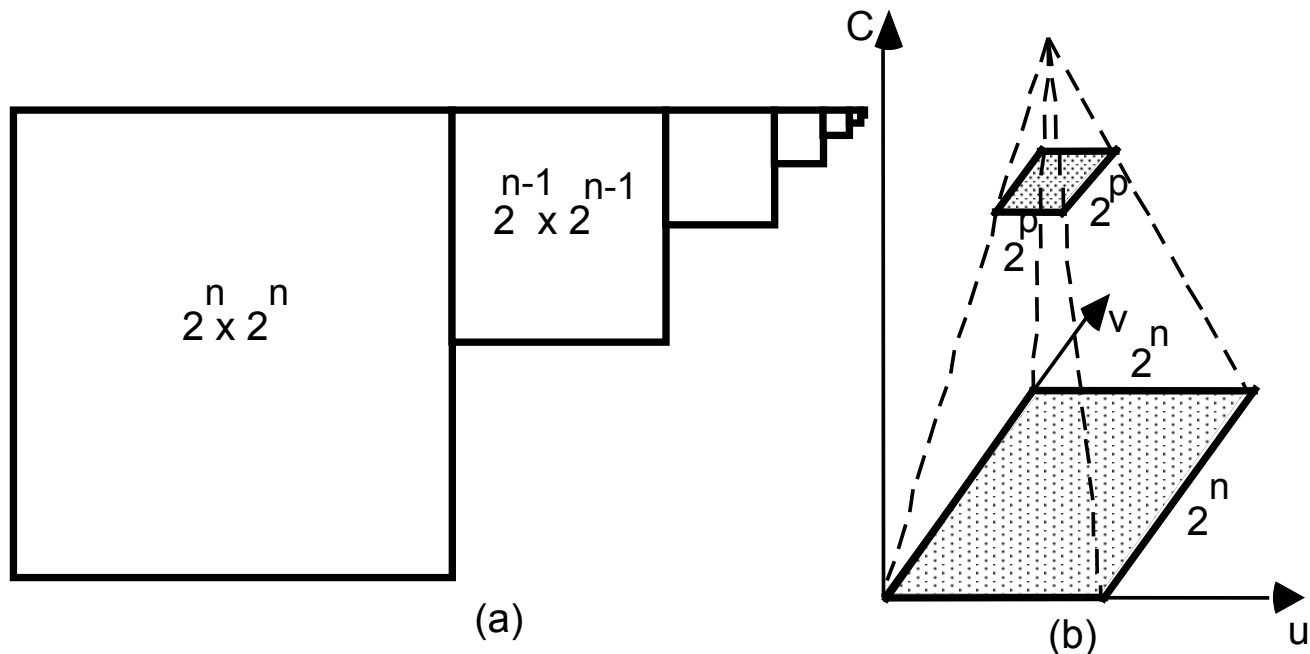
4 - Antialiasage

cause : compression du signal



méthodes de traitement

- filtrage simultané à la transformation perspective
- filtrage a priori (mip mapping MIP = multum in parvo)



➤ filtrage passe-bas

