Université de Liège

Infographie

Plan du cours

- Introduction
- Images et techniques d'affichage
  - Bases
  - Correction gamma
  - Aliasing et techniques pour y remédier
  - Stockage

Infographie

Plan du cours

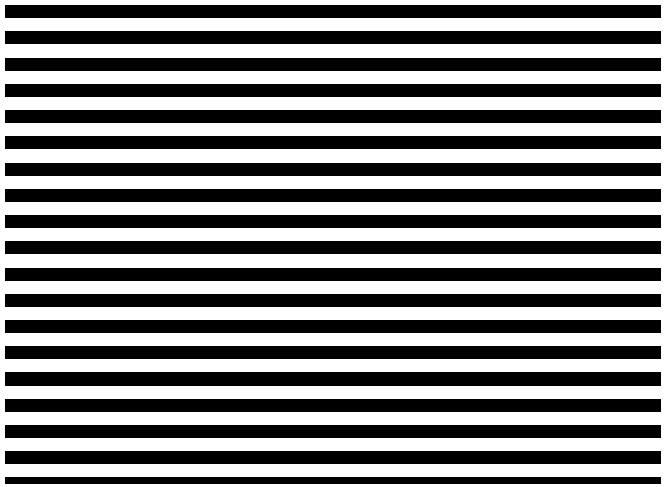
- Perspective 3D & transformations 2D / 3D
  - Passer d'un espace 3D vers un dispositif d'affichage 2D
- Deux paradigmes de génération d'images
- Représentation de courbes et surfaces
  - Splines & co.
  - Maillages
- Rendu réaliste (Ray tracing)
  - Concepts et bases théoriques

Infographie

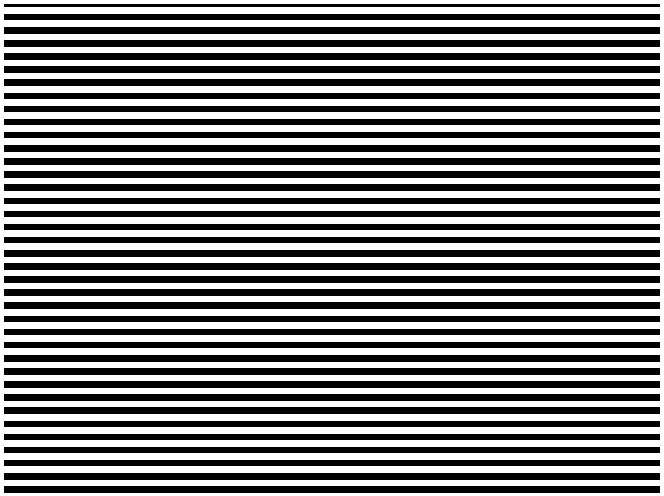
Aliasing

« Aliasing » (crénelage en français)

- Apparaît lors de la création ou la capture d'une image, lors de la discrétisation spatiale (transformation en pixels *discrets*)
  - Petite expérience : L'image à capturer est une succession de lignes noires et blanches dont la densité va augmentant.



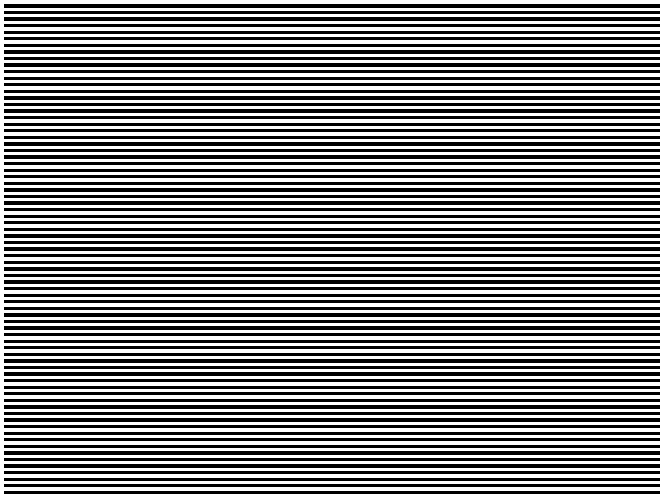
1 ligne/4 pixels



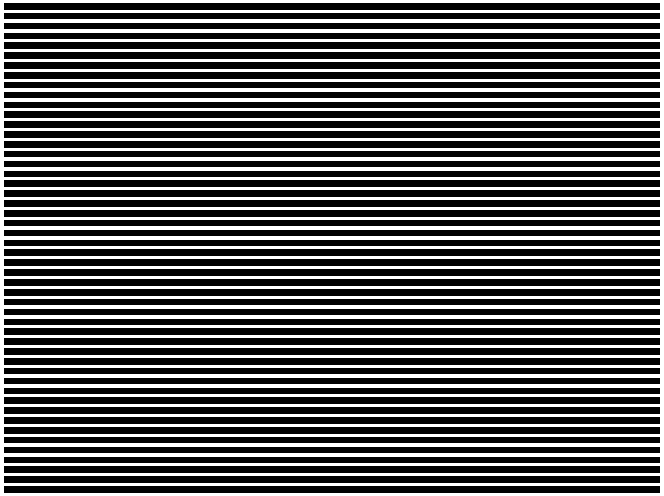
1/2

Infographie

Aliasing



1/1



3/2

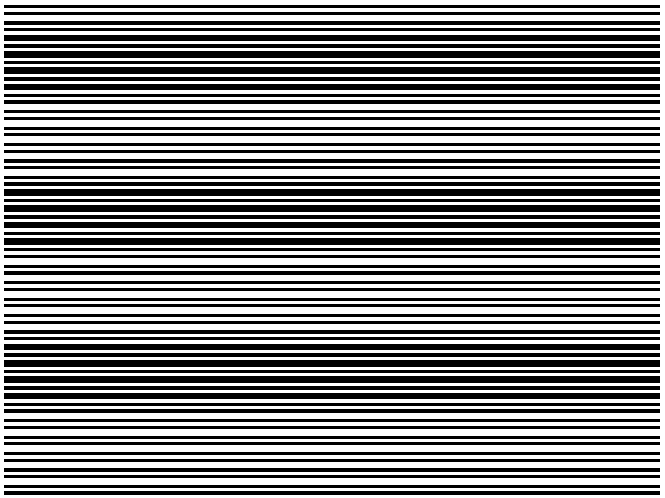
Infographie

Aliasing

2/1

199/100

Aliasing



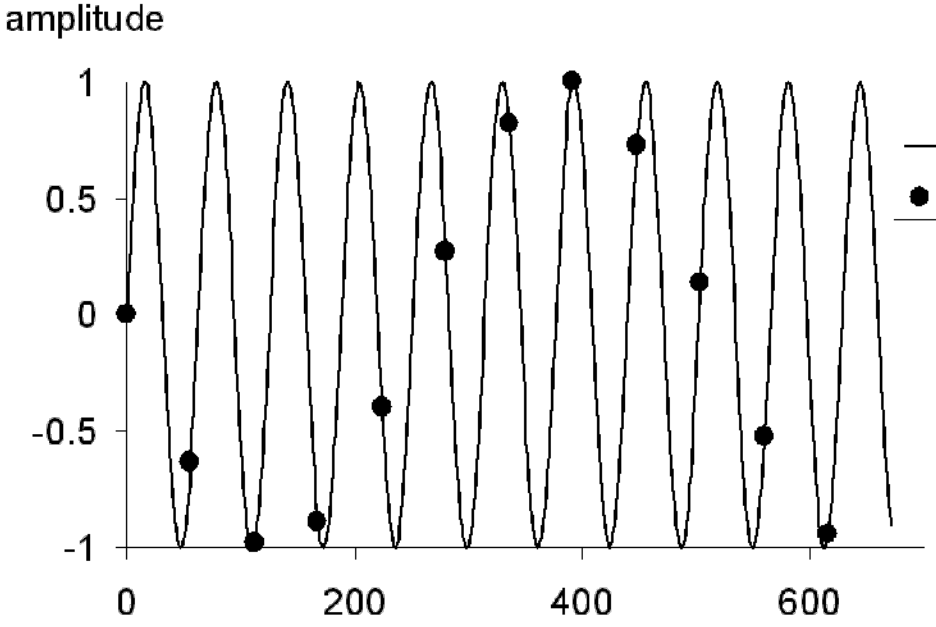
16/5

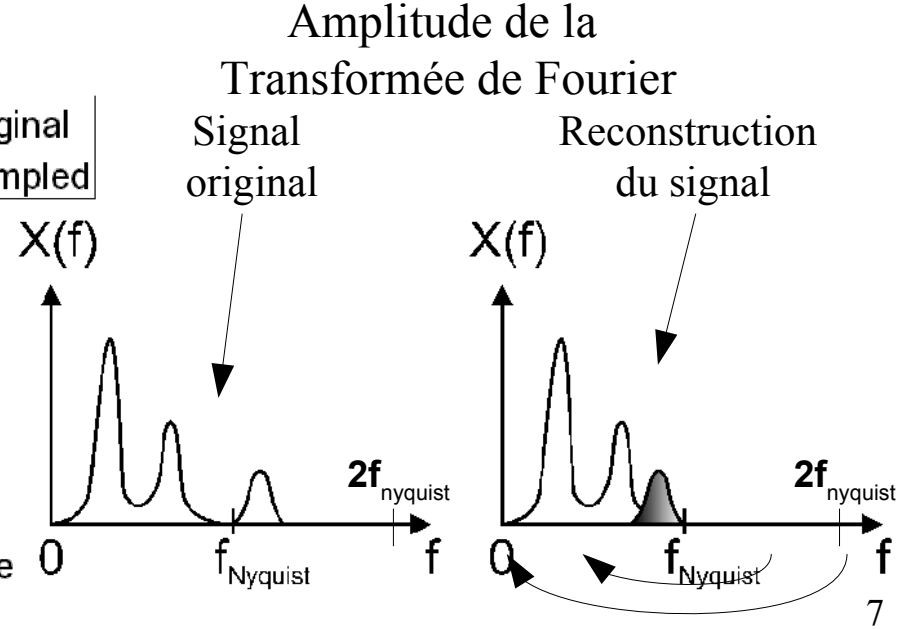
- En fait on ne devrait pas trouver de signal dont la fréquence spatiale est supérieure à une certaine constante reliée au pas d'échantillonage.

Infographie

Aliasing

- Théorème de Nyquist-Shannon
  - Si le spectre d'une fonction ne contient pas de fréquences supérieures à B, alors elle est complètement déterminée par une série d'échantillons séparés par 1/(2B) (en temps, espace, ...), soit de fréquence 2B. 2B est la fréquence de Nyquist.
- Dans le cas contraire, on a le phénomène d'aliasing :



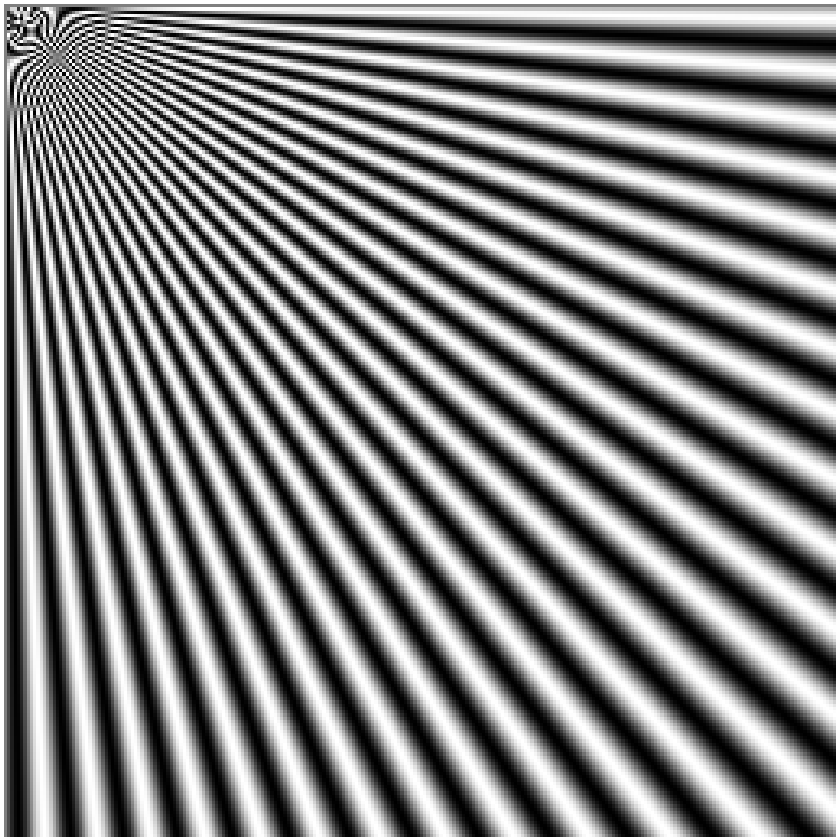


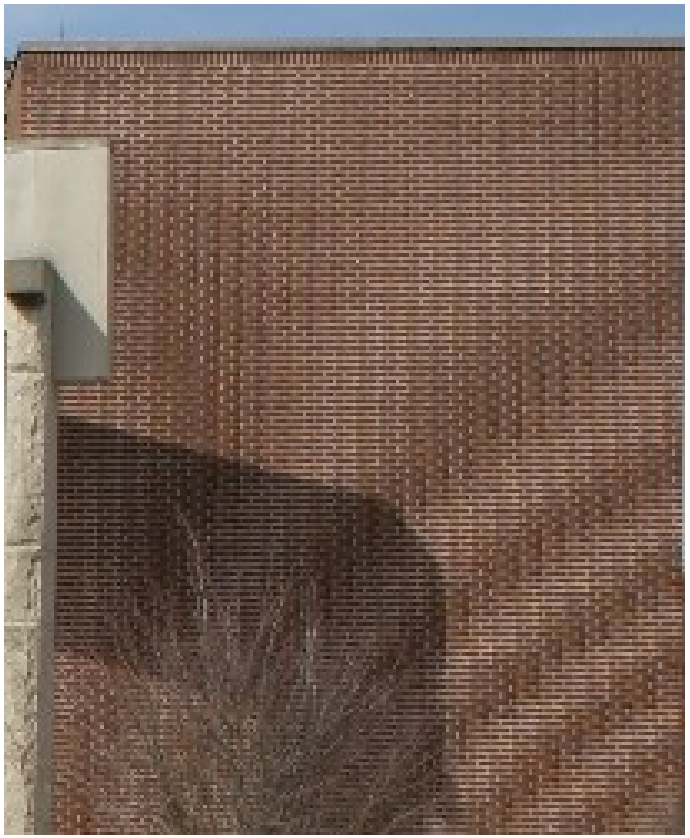
- Transformée de Fourier :

$$\hat{f}(\omega) = \int_{-\infty}^{+\infty} f(t)\, e^{i\omega t}\, dt$$

Aliasing

- Phénomène de moiré





Aliasing

Comment supprimer l'aliasing ?

- En augmentant la densité de l'échantillonnage jusqu'à respecter le théorème de Shannon ?
- En filtrant le signal dans un filtre passe bas afin d'être dans le fenêtre du théorème de Shannon pour une densité d'échantillons donnée ?
- Une combinaison des deux... ?

Infographie

Aliasing

- Augmenter la densité des échantillons
  - Revient à accroître la résolution de l'image
  - Mais : les images réelles sont souvent d'aspect fractal
    - La fréquence (résolution spatiale) des détails est énorme
  - Mais : les images générées artificiellement comportent de brusques variations d'intensité (ligne noire sur fond blanc)
    - Cf décomposition de Fourier d'un signal carré :

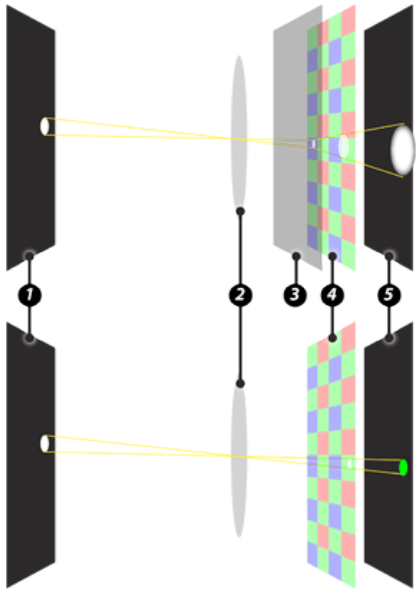
      Spectre du signal carré : harmoniques impaires d'amplitude 1/n ...

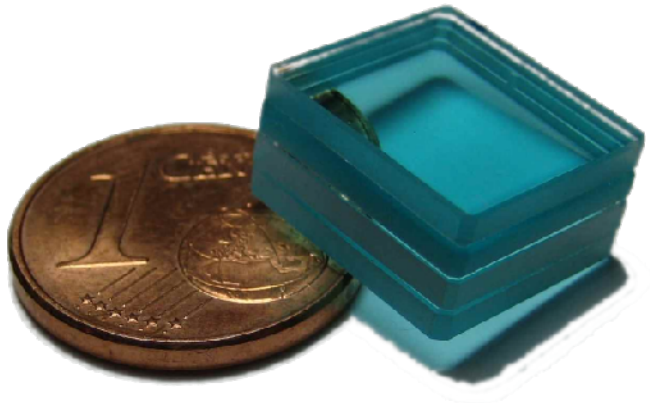
Seule, cette solution ne suffit pas.

Infographie

Aliasing

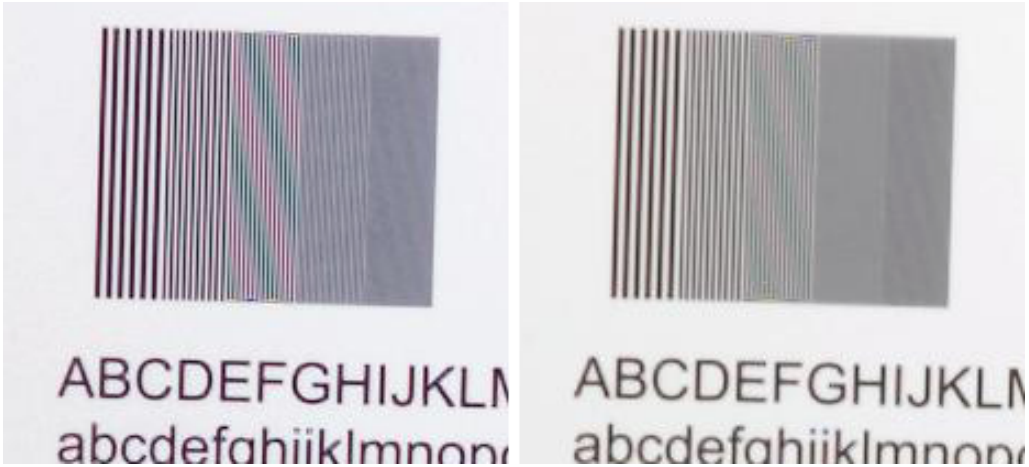
- Filtrage passe bas « analogique »
  - Pour les dispositifs de capture, ceci doit être fait avant la capture proprement dite
  - Revient à rendre l'image légèrement floue – mais pas trop !
  - C'est exactement ce que l'on trouve sur le capteur des appareils photos numériques. Un filtre positionné entre l'objectif et le capteur.





- Exemple : filtre analogique (physique) dans un appareil photo numérique

Sans filtre antialiasing



Avec filtre AA devant le capteur

Infographie

Aliasing

- Filtrage passe bas numérique
  - Lors de la création d'images de synthèse, on doit faire autrement !
  - Une manière courante est de faire de l'oversampling (c'est à dire échantillonner à plus haute fréquence et faire une moyenne des échantillons lors du retour à la fréquence voulue

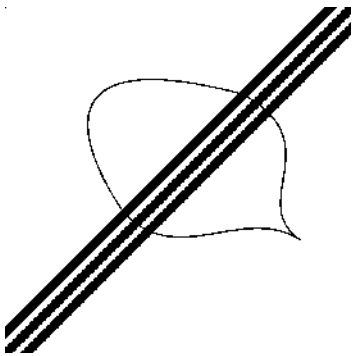
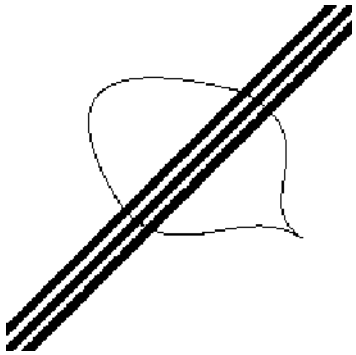
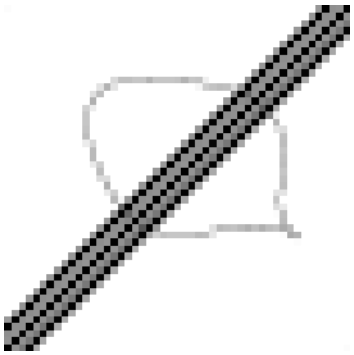


Image originale (continue)

Résolution 49x49 Sans antialiasing



Résolution 196x196 Sans antialiasing



Résolution 49x49 avec oversampling 4x4

Infographie

Aliasing

- Simulation d'un filtre passe bas analogique

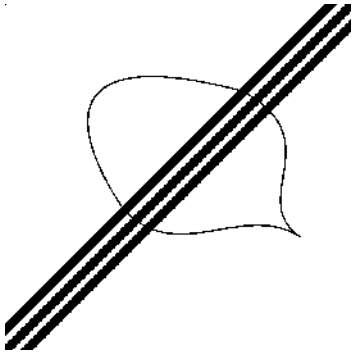
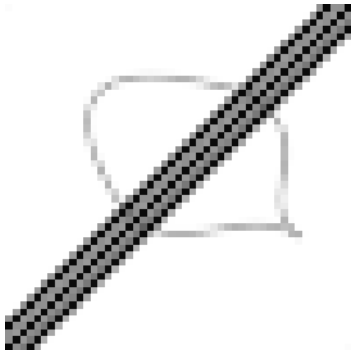
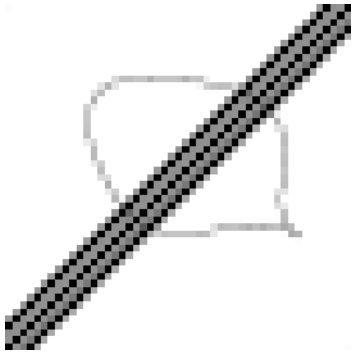


Image originale (continue)

Image originale « floutée »



Résolution 49x49 à partir de l'image « floutée »



Pour rappel Résolution 49x49 avec oversampling 4x4

Infographie

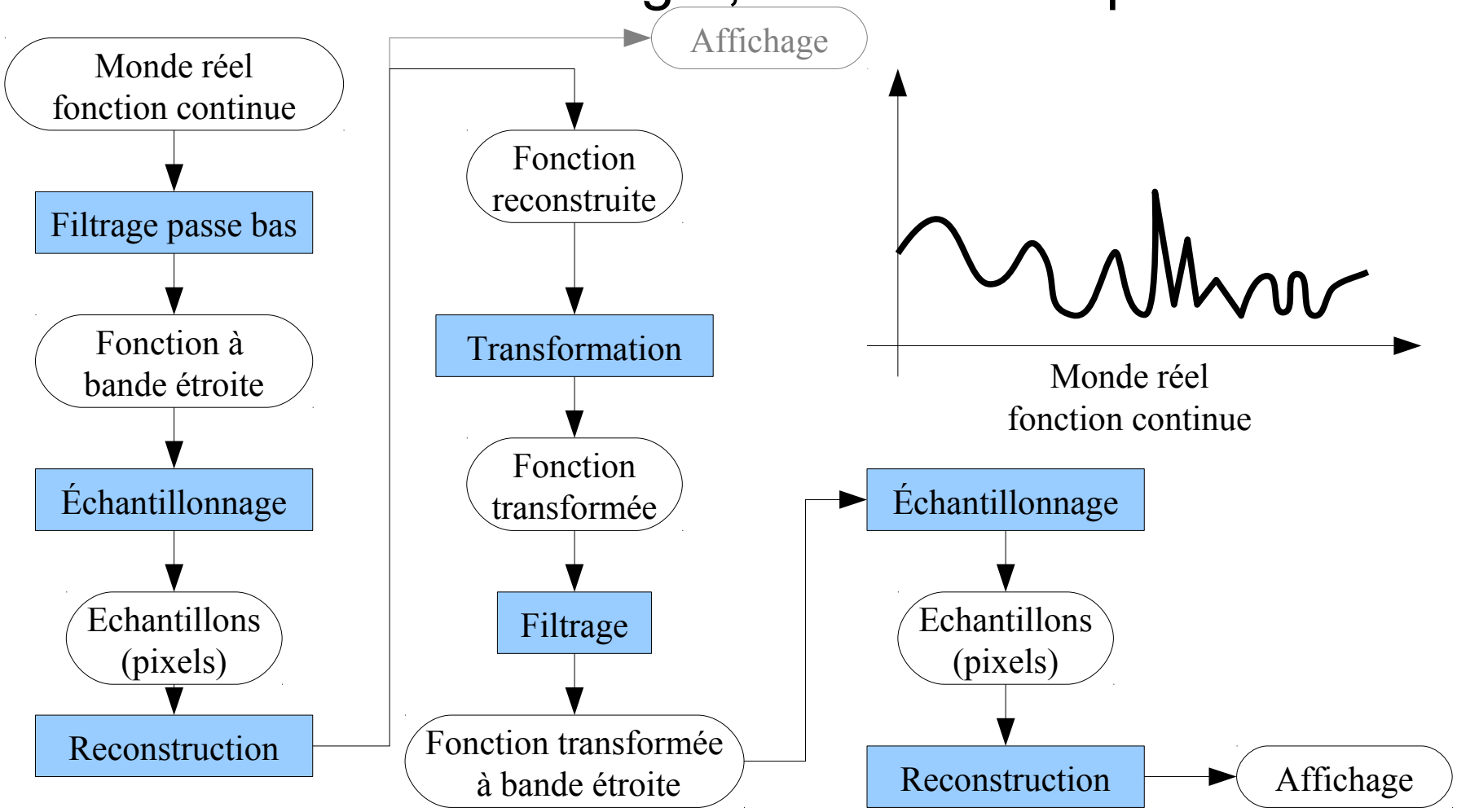
Aliasing

- Filtres habituels utilisés lors du rééchantillonnage d'une image
  - Plus proche voisin
    - Fort aliasing
  - Interpolation bilinéaire (4 plus proches pixels)
    - Apparence plus douce
  - Interpolation bicubique (16 plus proches pixels)
  - Filtre gaussien
  - Lanczos
    - Utilisation d'une approximation de la fonction sinc (filtre passe bas idéal)

Infographie

Aliasing

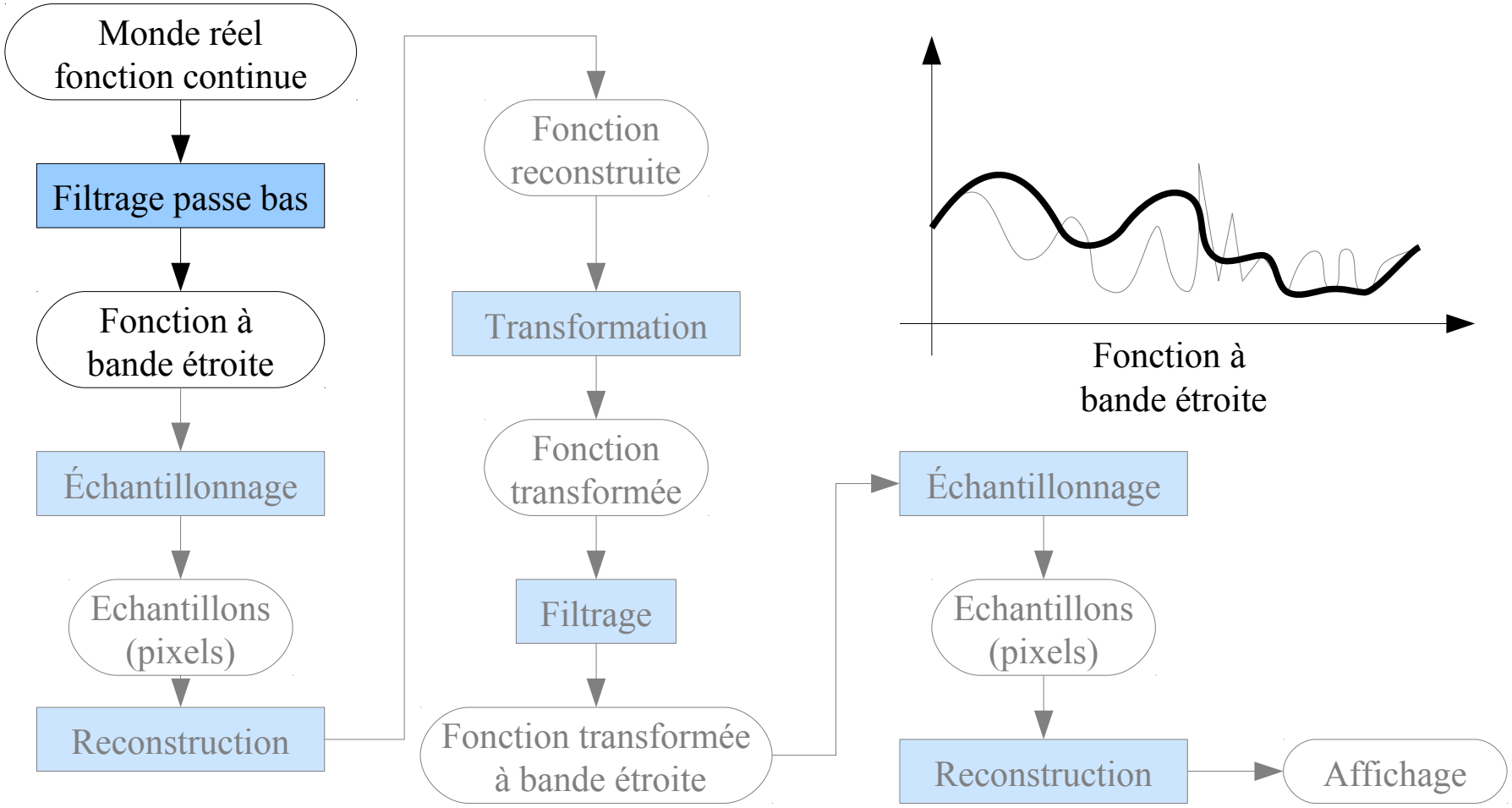
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

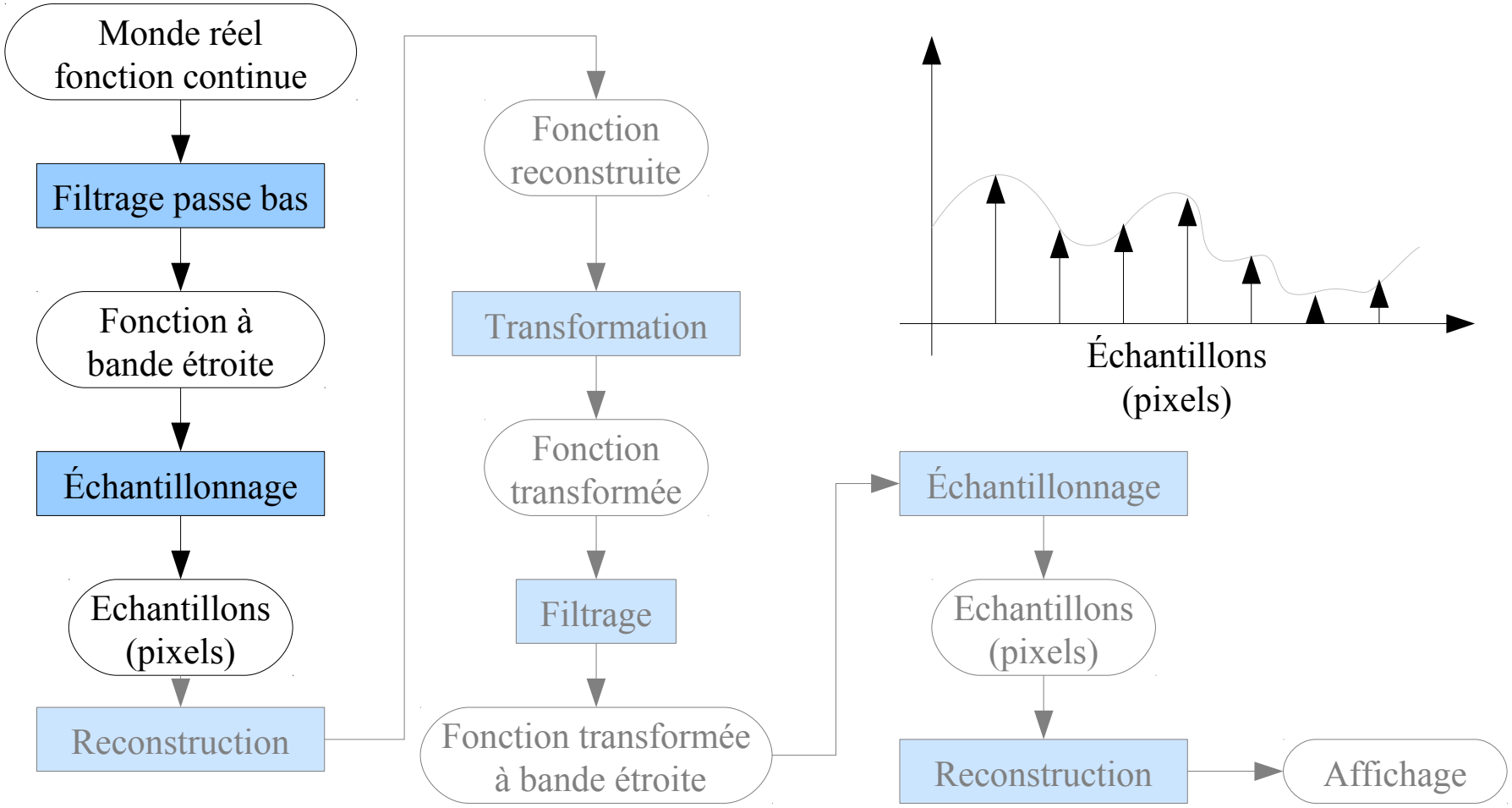
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

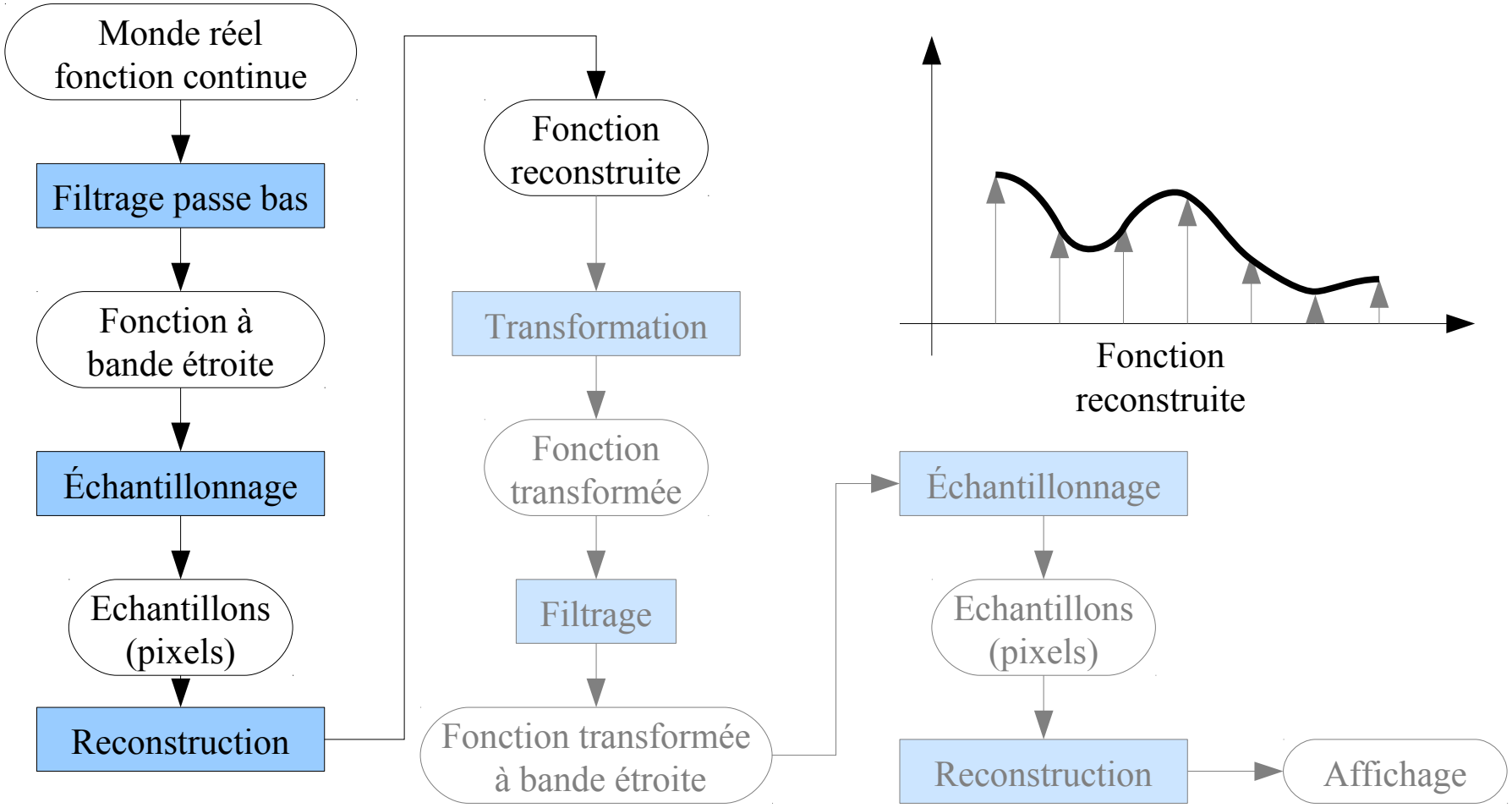
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

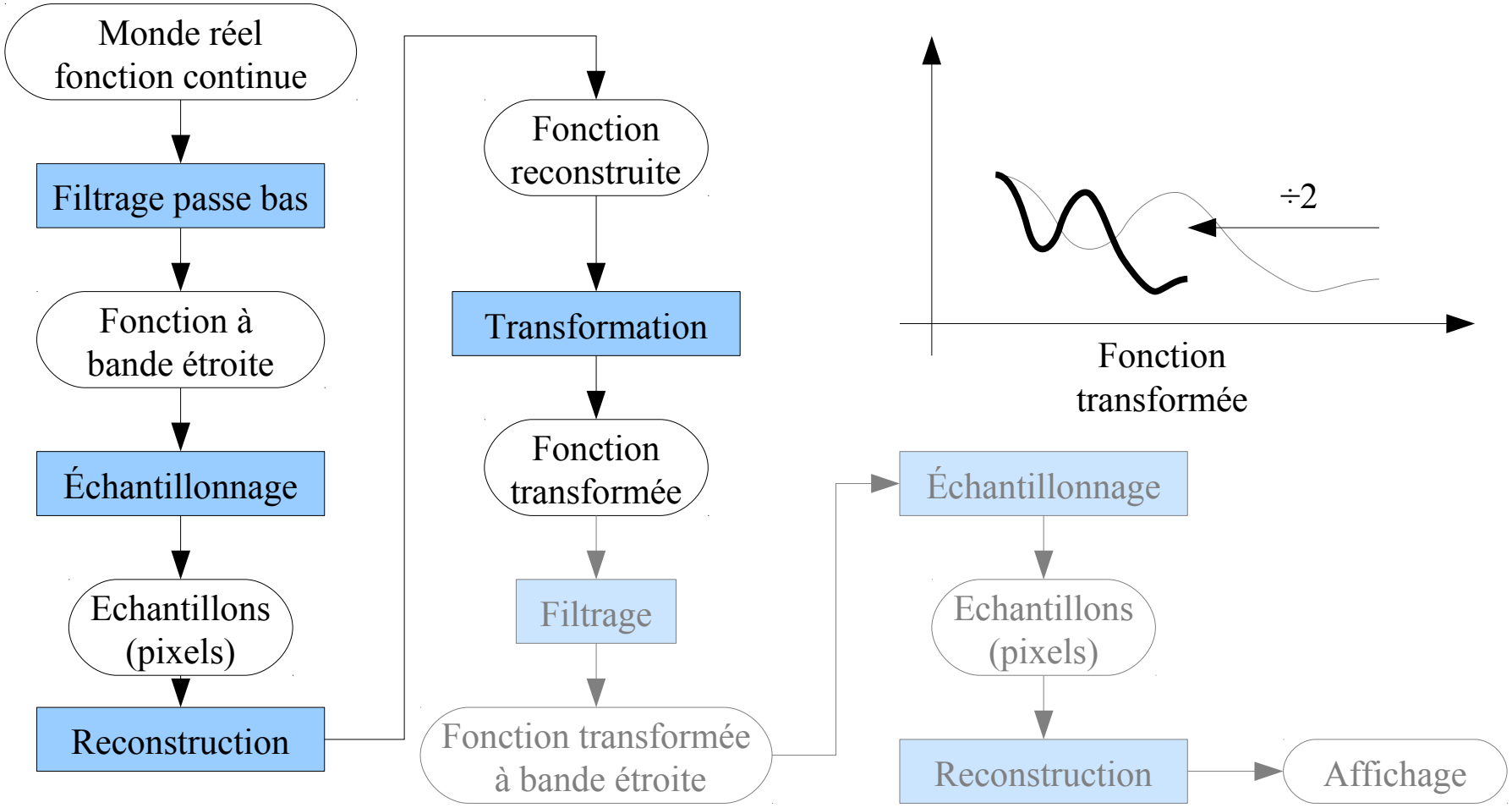
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

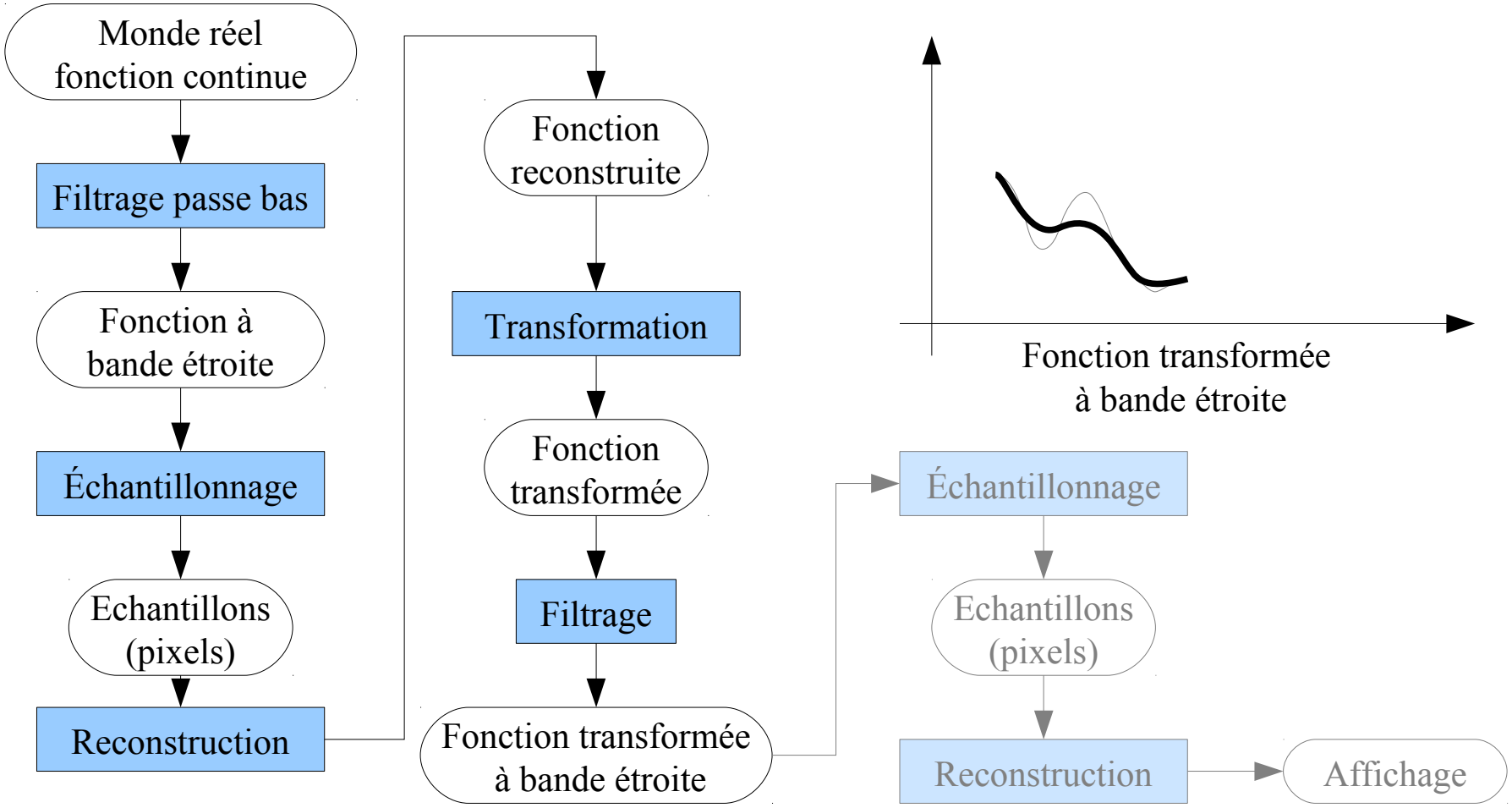
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

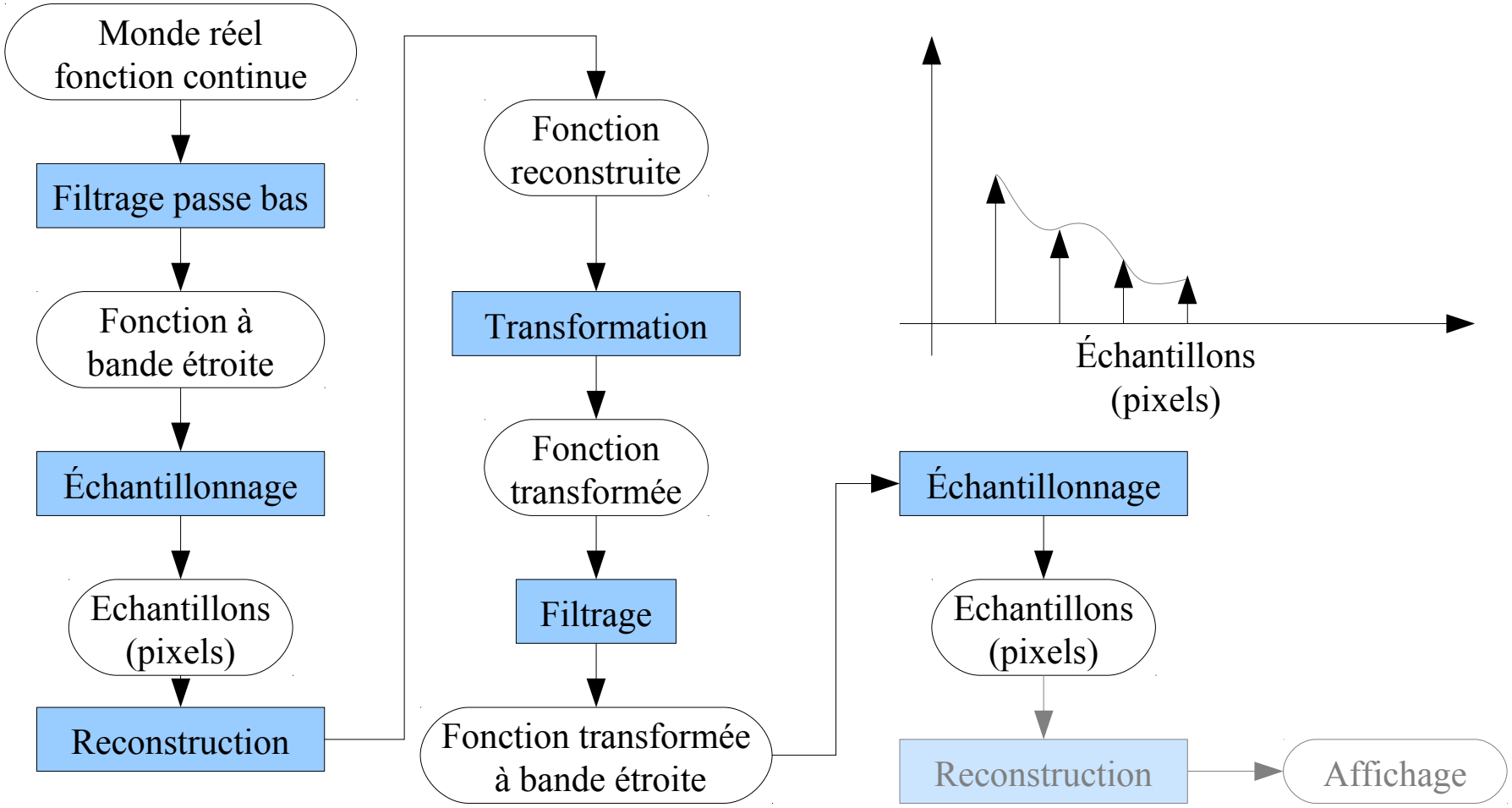
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

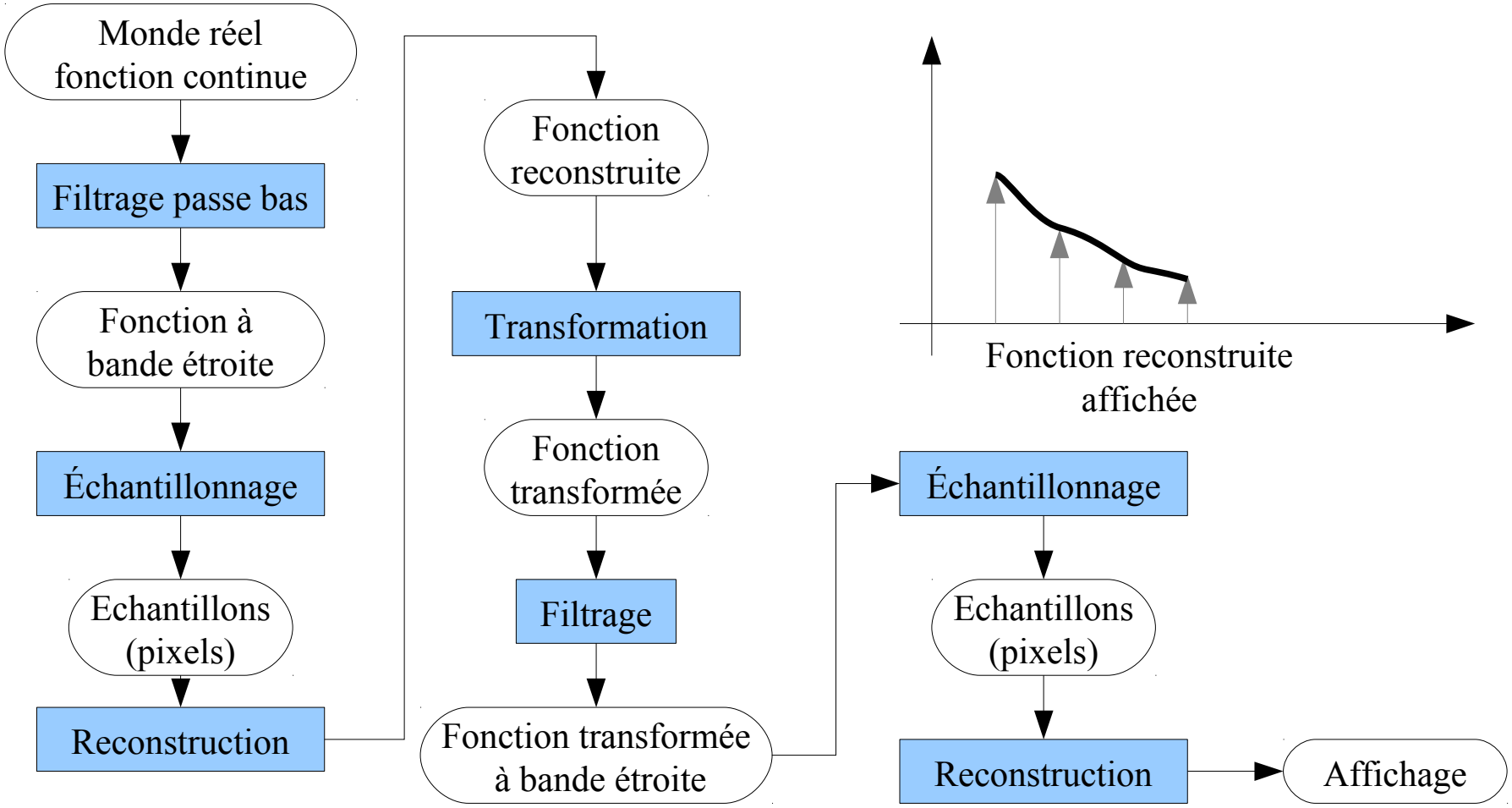
- Traitement de l'image ; une série d'opérations



Infographie

Aliasing

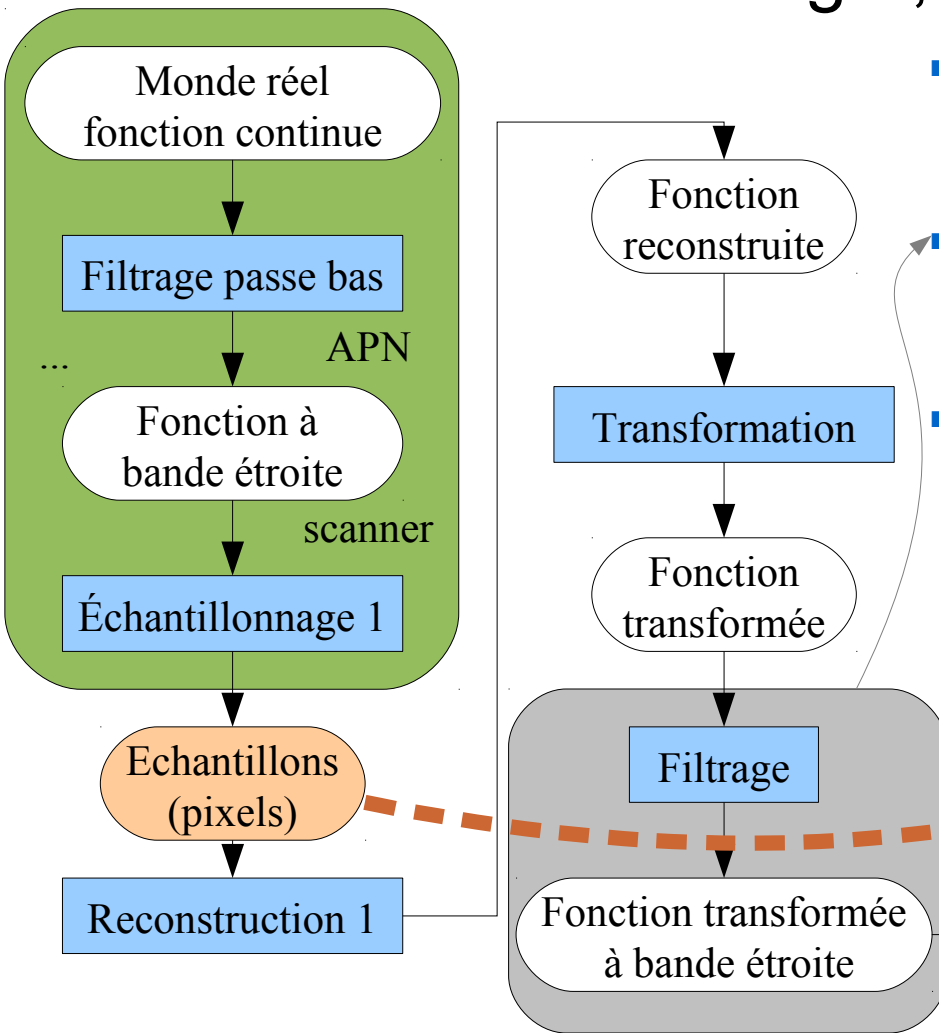
- Traitement de l'image ; une série d'opérations

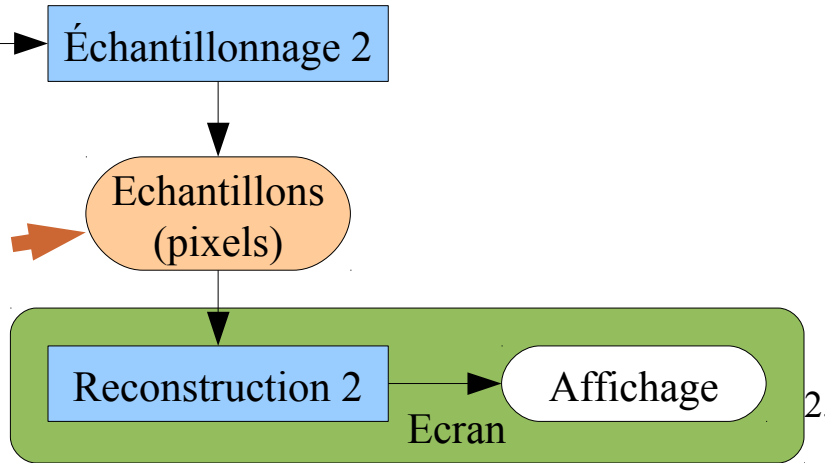


Infographie

Aliasing

- Traitement de l'image ; une série d'opérations



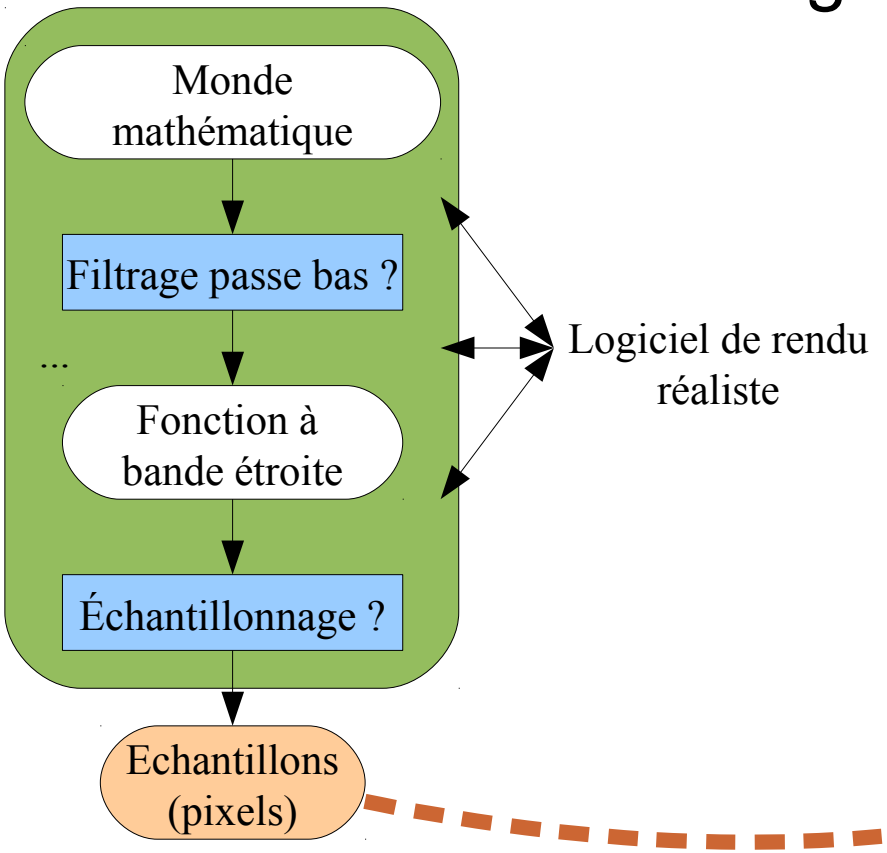


- Échantillonnage 1 – dans un appareil ...
- Filtrage optionnel
  - Peut être inclus dans la reconstr. 1
- Reconstruction 2 – dans un appareil (écran!)

Infographie

Aliasing

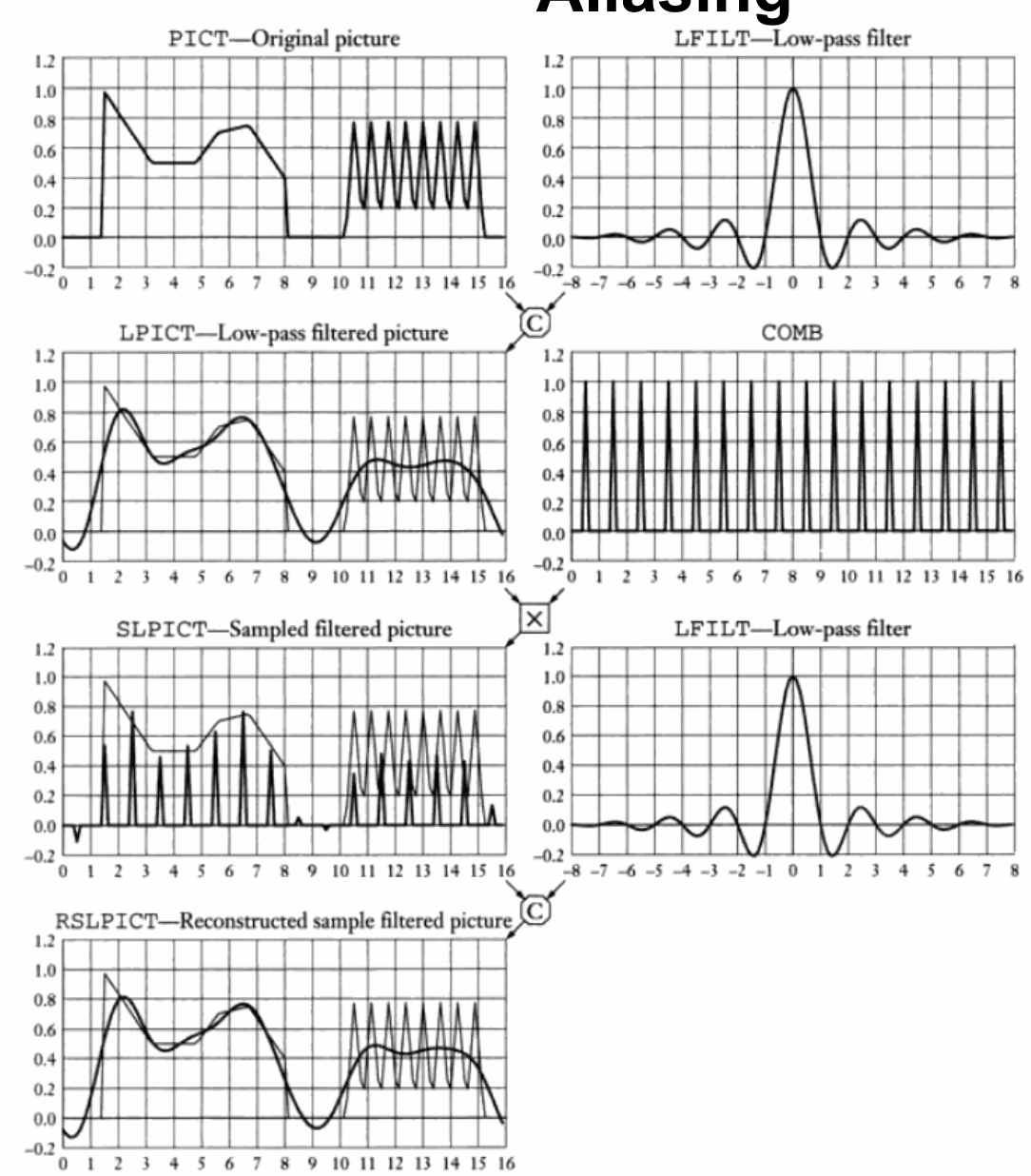
- Traitement de l'image : cas des images de synthèse



- La question se pose à propose du filtrage passe-bas et de l'échantillonnage du monde « mathématique » que l'on souhaite représenter.
- Difficile de séparer les deux opérations
- J'en reparlerai plus tard.

Infographie

Aliasing

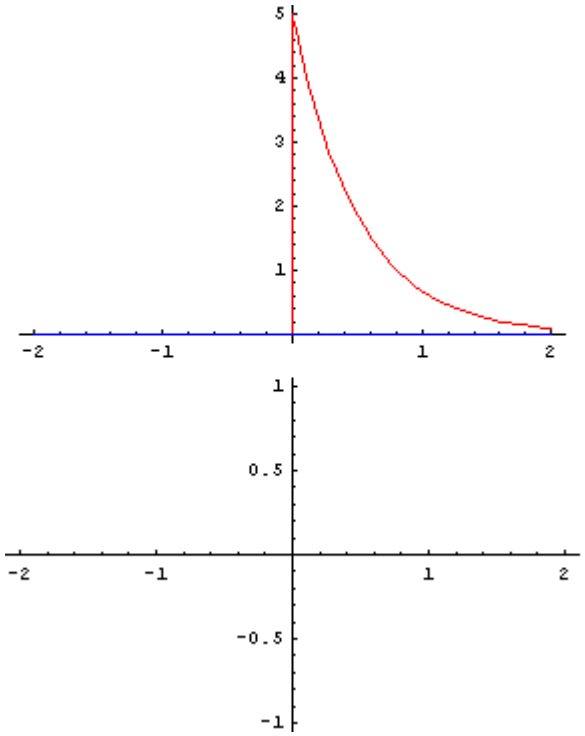


Infographie

Aliasing

- Convolution

$$(f * g)(t) = \int_{-\infty}^{+\infty} f(t-\tau)\, g(\tau)\, d\tau$$

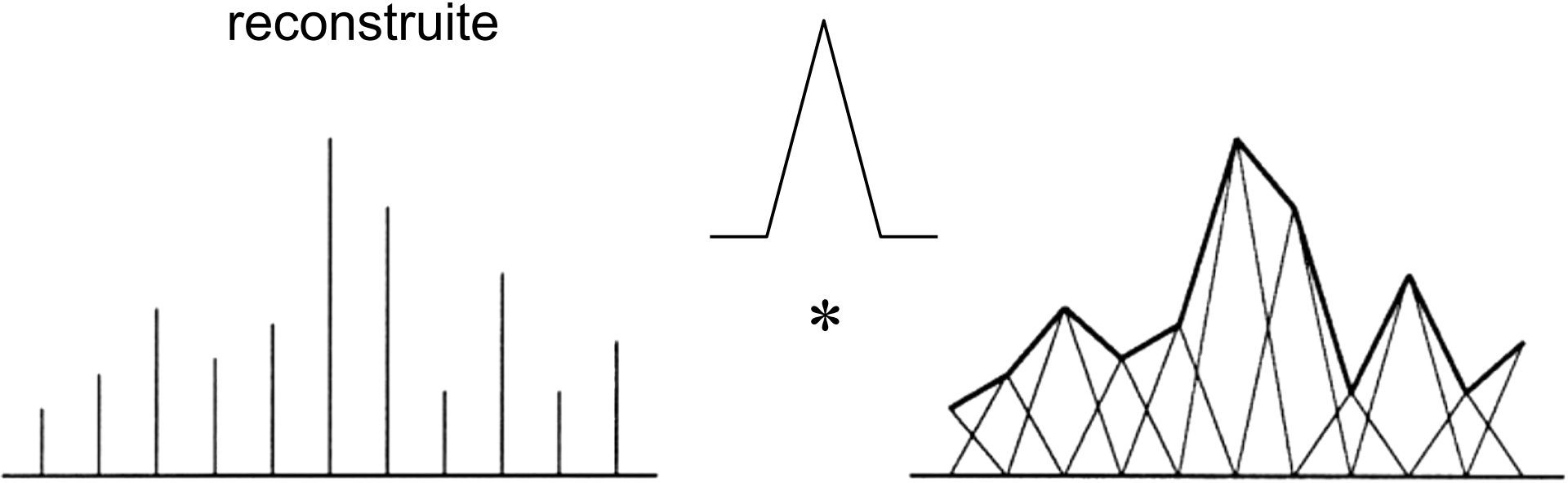


Aliasing

- Reconstruction
  - Partir des échantillons
  - Effectuer une convolution avec un certaine fonction
    - Linéaire, bicubique, Gaussien, etc...
    - Revient à effectuer une interpolation pour reconstruire le signal là ou il n'existe plus (entre les échantillons)
    - Permet donc de revenir à un signal continu

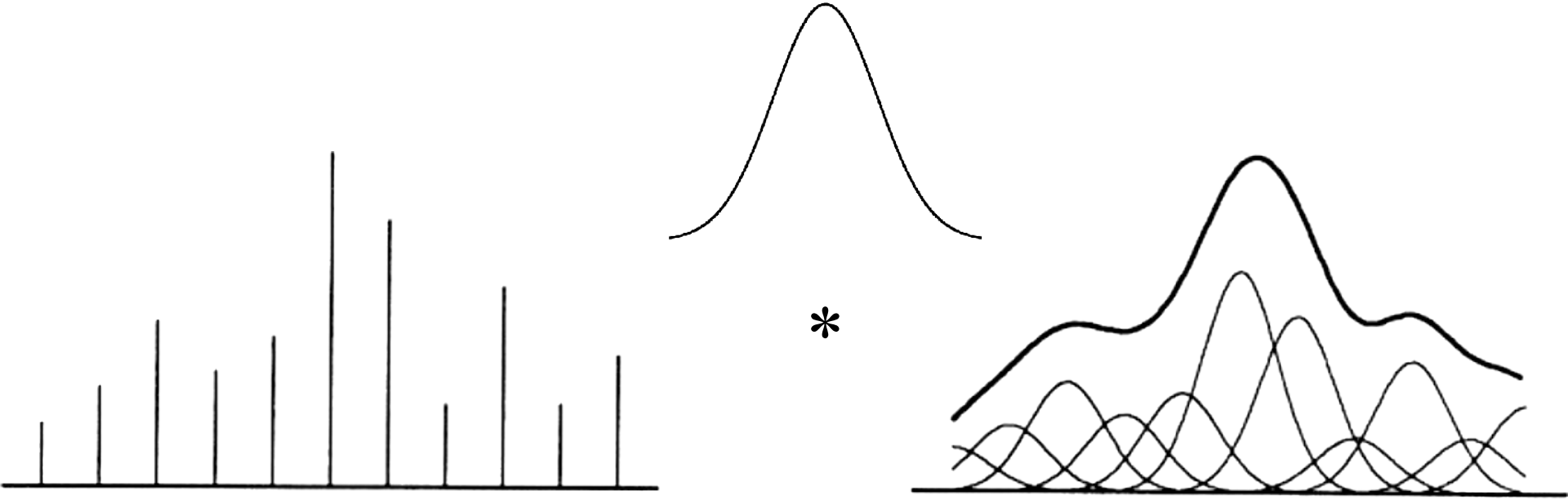
Aliasing

- Reconstruction avec un filtre linéaire (« chapeau »)
  - Chaque échantillon est « multiplié » par la fonction chapeau et la somme constitue la fonction reconstruite



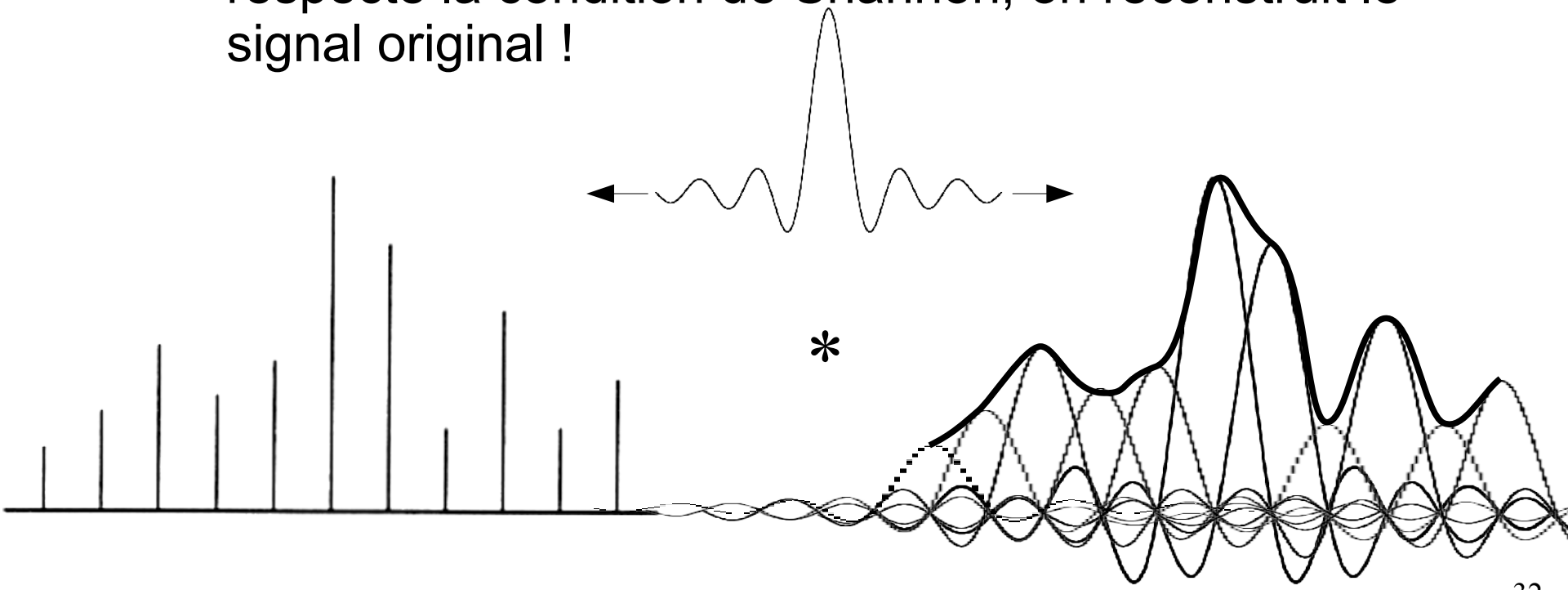
Aliasing

- Reconstruction avec un filtre gaussien
  - Chaque échantillon est « multiplié » par la fonction gaussienne et la somme constitue la fonction reconstruite



Aliasing

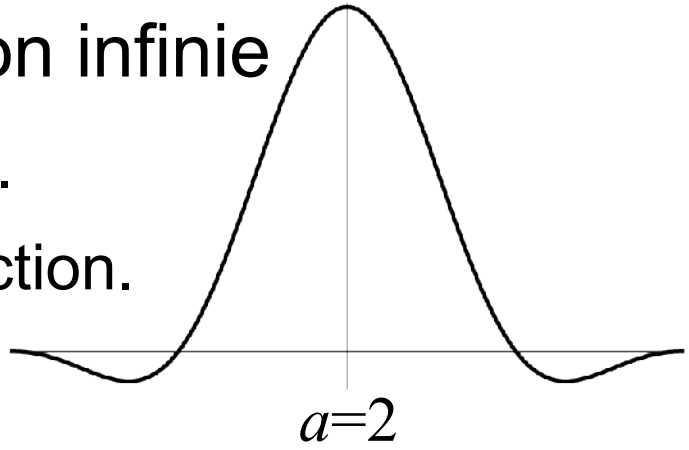
- Reconstruction avec sinc $sinc(x) = \frac{\sin \pi x}{\pi x}$
  - On fait la convolution avec un sinus cardinal - extension infinie : sous réserve que le signal original respecte la condition de Shannon, on reconstruit le signal original !



Infographie

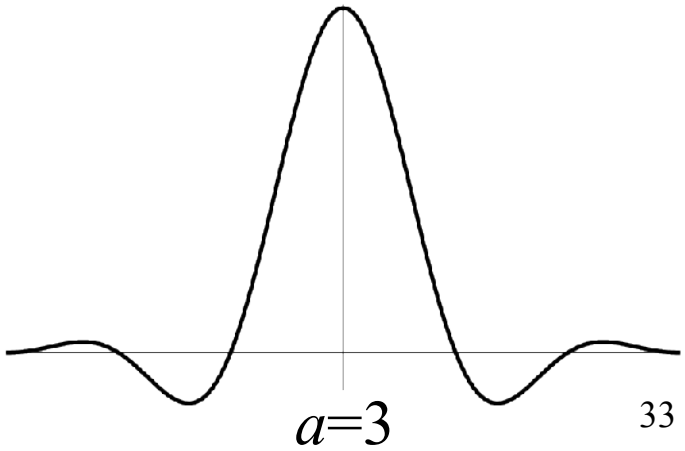
Aliasing

- Le sinus cardinal est d'extension infinie
  - En théorie, effort de calcul infini...
  - En pratique on tronque cette fonction.



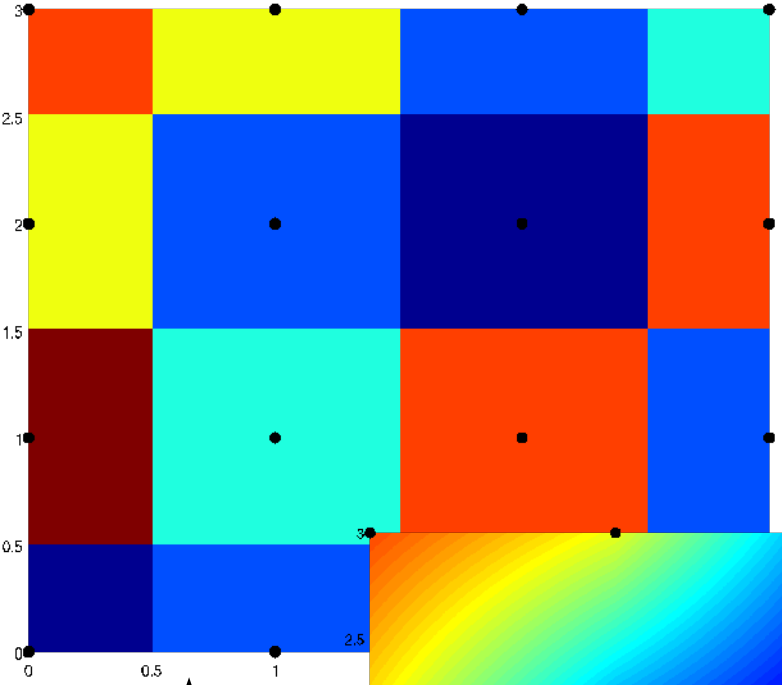
$$L(x)=\begin{cases} sinc(x)\,sinc(x/a) & \text{si} \quad -a<x<a\,,\ x\neq 0 \\ 1 & \text{si} \quad x=0 \\ 0 & \text{sinon} \end{cases}$$

- On obtient le filtre de Lanczos

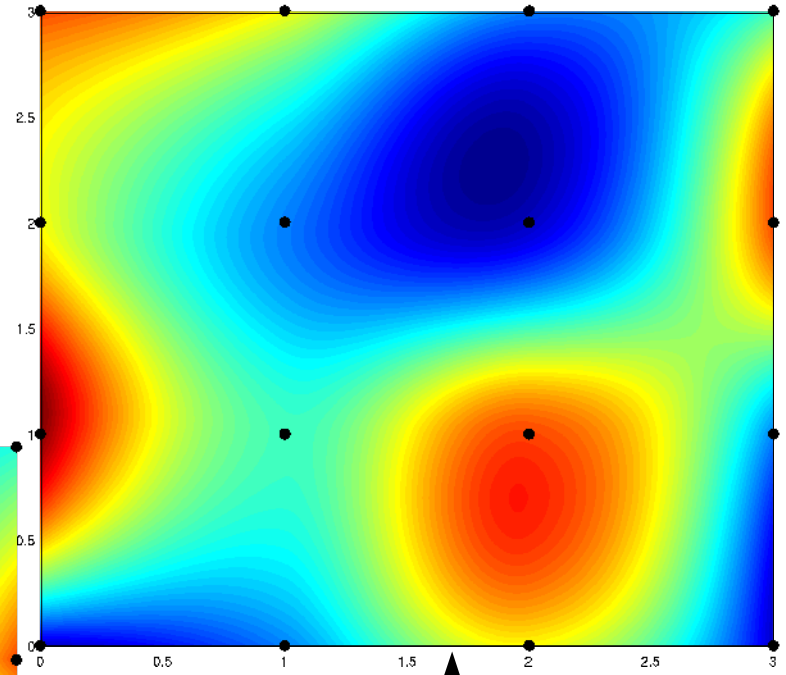


Infographie

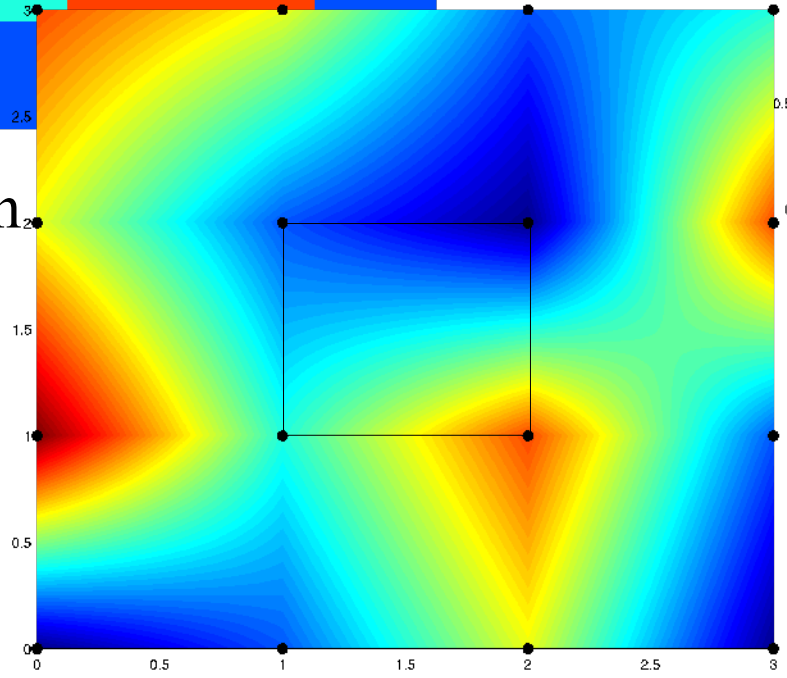
Aliasing



Interpolation plus proche voisin



Interpolation bicubique



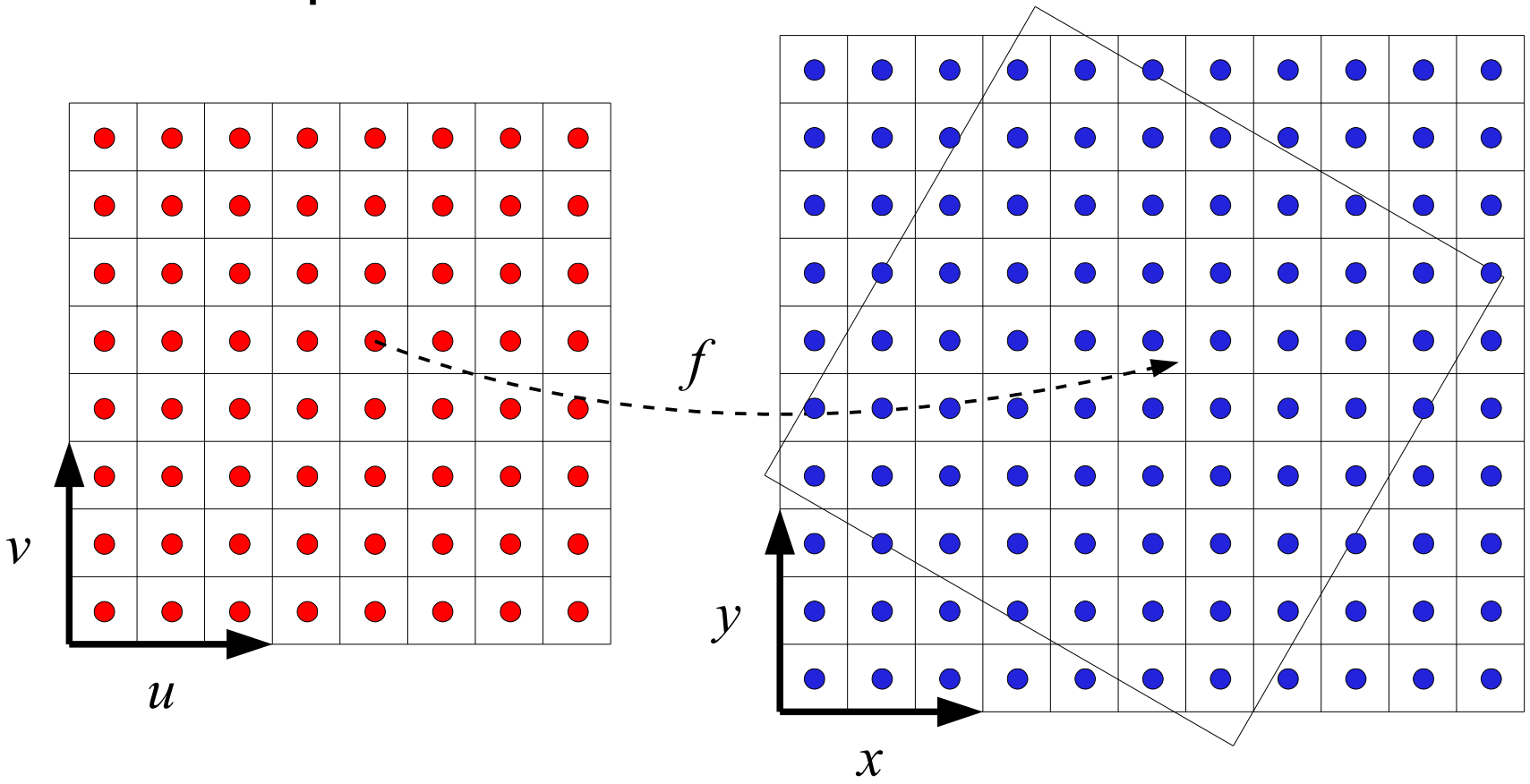
Interpolation bilinéaire

Aliasing

- Transformation
  - Consiste à modifier la position des échantillons de l'image
    - Rotation
    - Mise à l'échelle
    - Etc...
  - Dans le cas ou la discrétisation cible est inférieure, il faut à nouveau filtrer (passe bas) avant d'échantillonner
  - Si la discrétisation cible est identique ou plus fine, il suffit de ré-échantillonner après la reconstruction

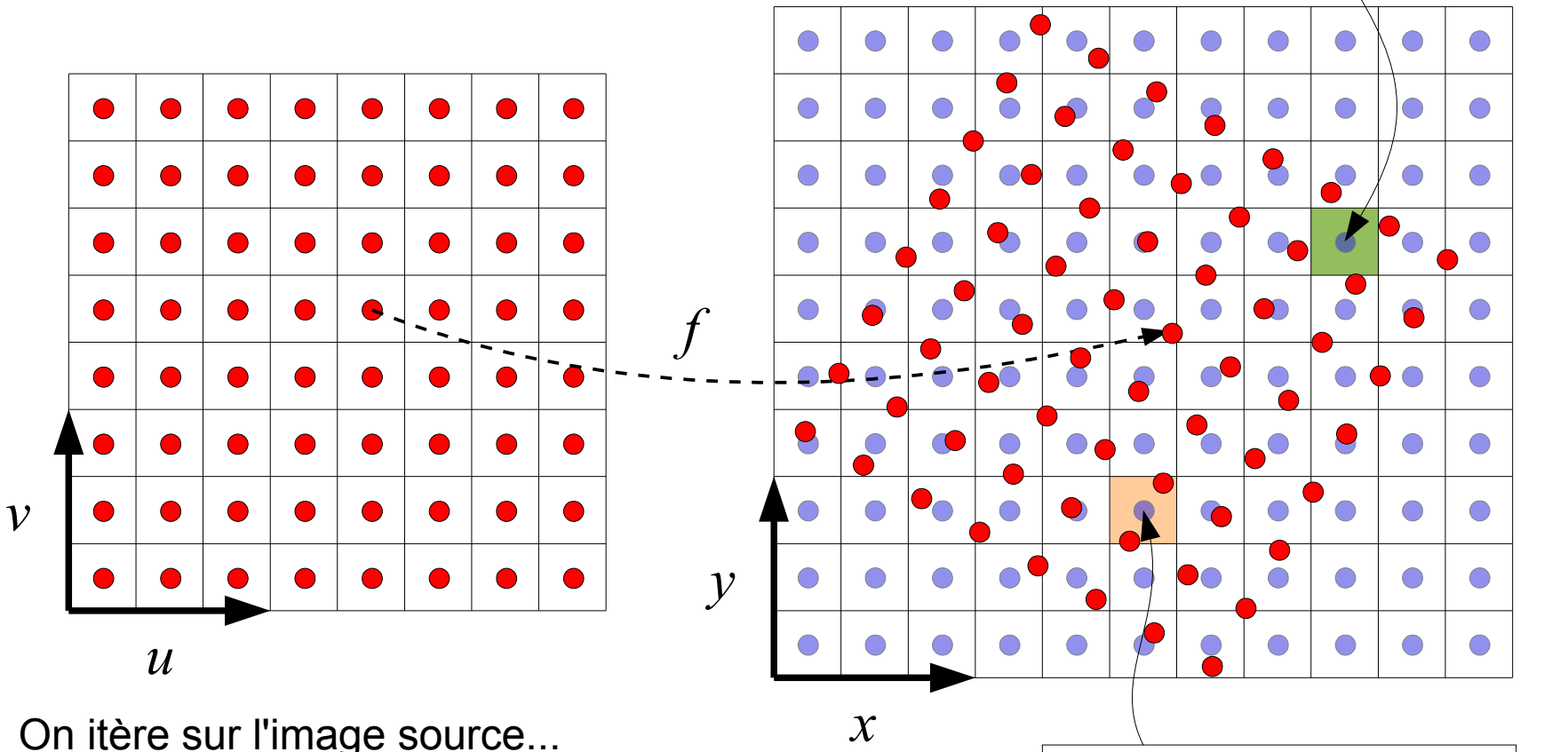
Aliasing

- Exemple avec une rotation



Aliasing

- Mapping direct – mauvaise idée

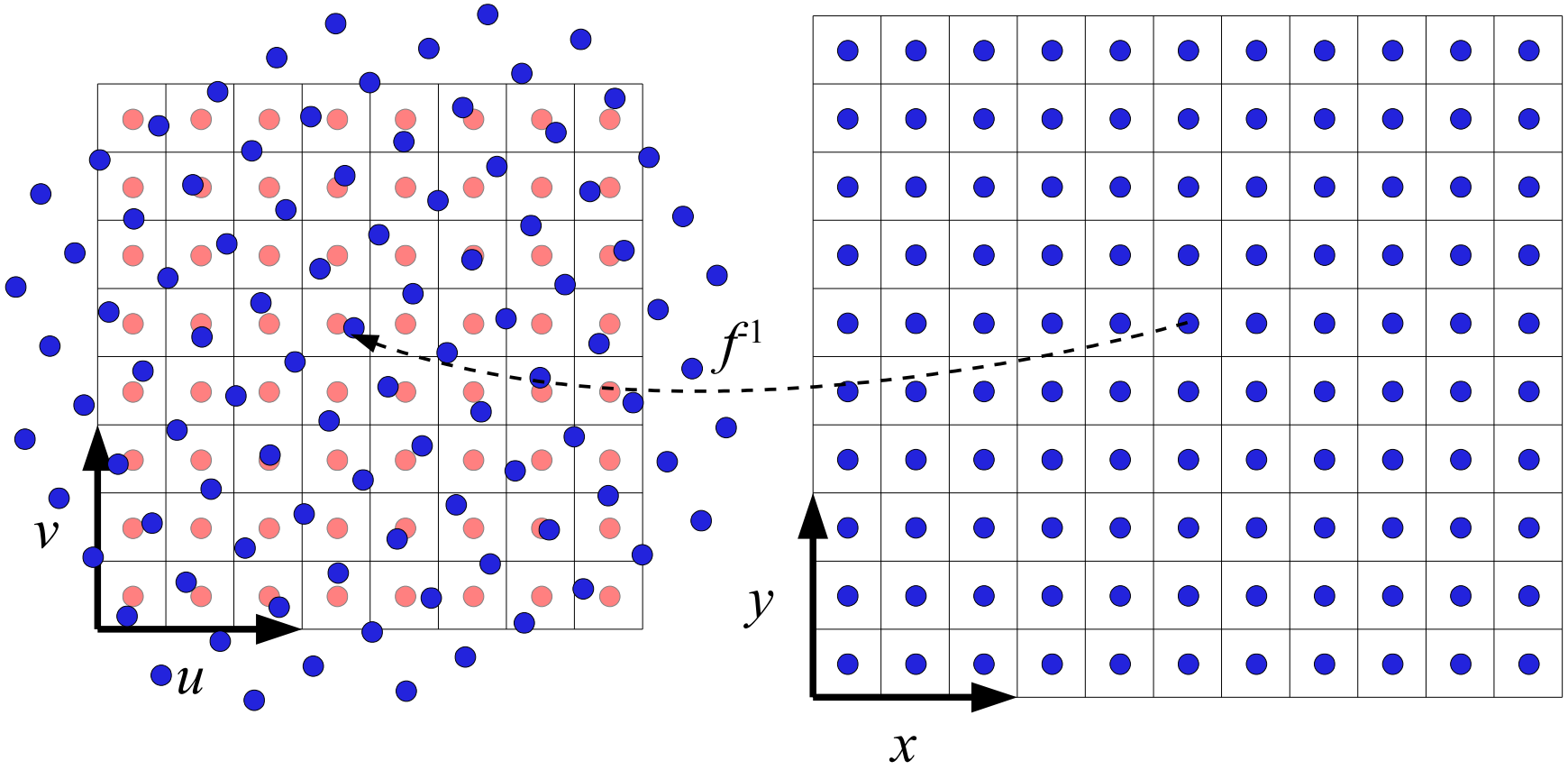


On itère sur l'image source...
mais impossible de travailler là ou il n'y a pas d'information dans l'image cible

Infographie

Aliasing

- Mapping inverse



On itère sur l'image cible ... - Il faut rééchantilloner
- On utilise une reconstruction de l'image source

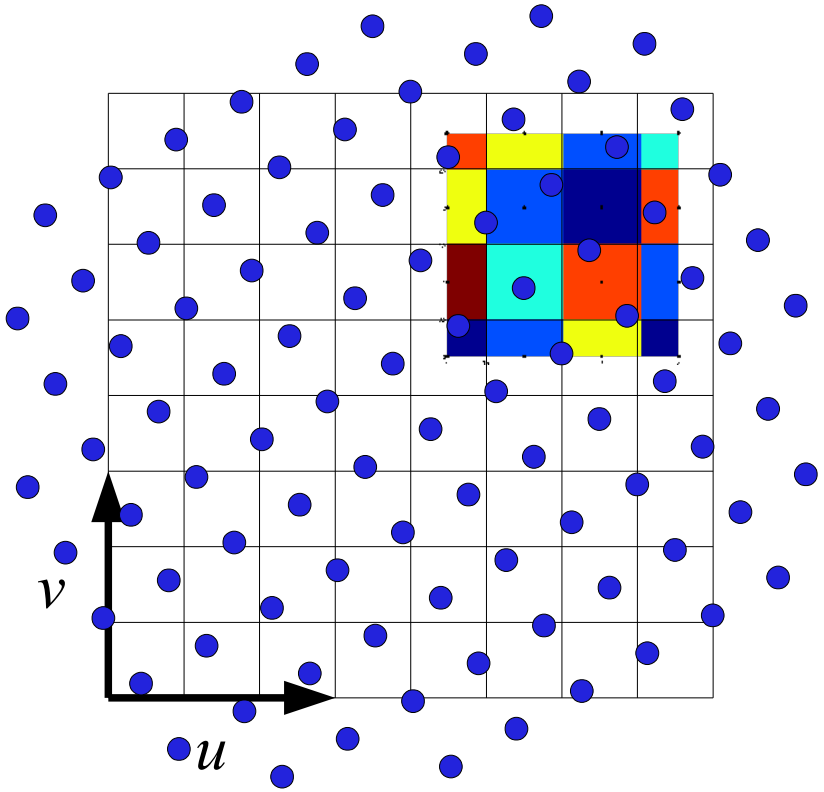
Aliasing

- La reconstruction est déterminante dans la qualité de l'image cible
  - Plus proche voisin – beaucoup d'aliasing
  - Bilinéaire – flou (perte de détails)
  - Bicubique et « Lanczos » - meilleurs mais coûteux

Infographie

Aliasing

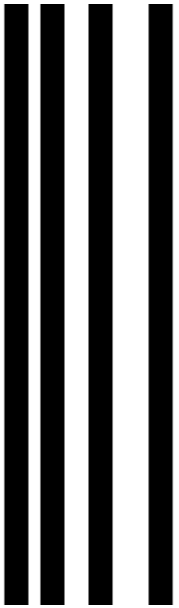
- Plus proche voisin :



Infographie

Aliasing

- Série de 36 rotations de 5° -> 180° suivi d'un retournement (sans pertes)
  - Images originales

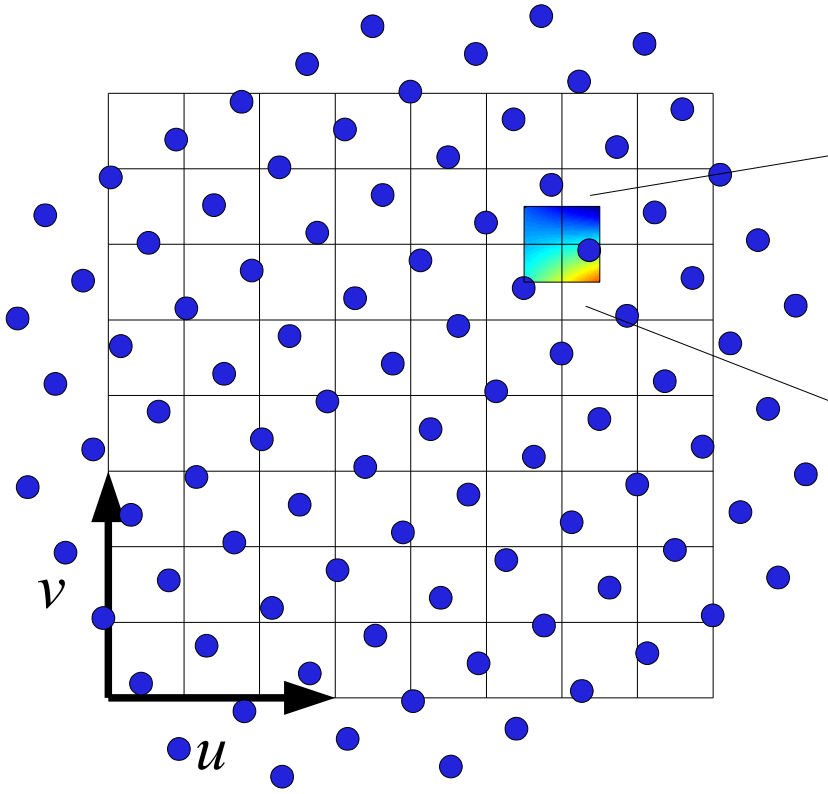


- Filtre « plus proche voisin »

Infographie

Aliasing

- Bilinéaire :



$f(0,1)$ $f(1,1)$

$\bar{v} = \frac{v - v_{min}}{v_{max} - v_{min}}$

$f(\bar{u},\bar{v})$

$f(0,0)$ $f(1,0)$

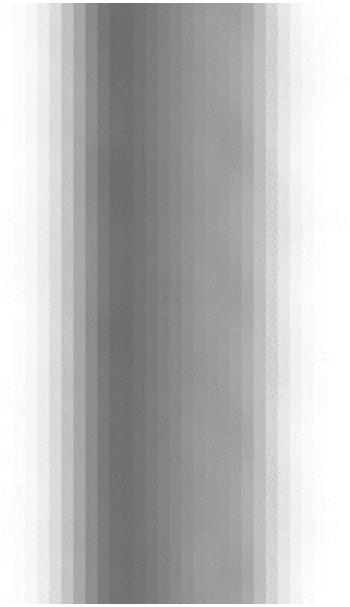
$\bar{u} = \frac{u - u_{min}}{u_{max} - u_{min}}$

$$f(\bar{u},\bar{v}) = f(0,0)(1-\bar{u})(1-\bar{v}) + f(1,0)\bar{u}(1-\bar{v}) + f(0,1)(1-\bar{u})\bar{v} + f(1,1)\bar{u}\bar{v}$$

Infographie

Aliasing

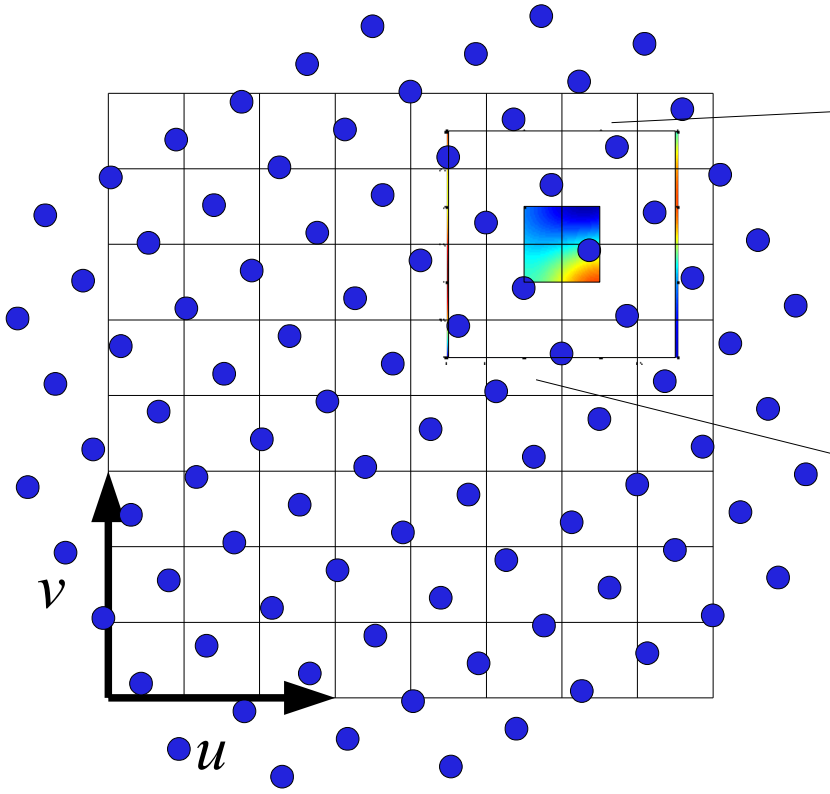
- Filtre « bilinéaire »



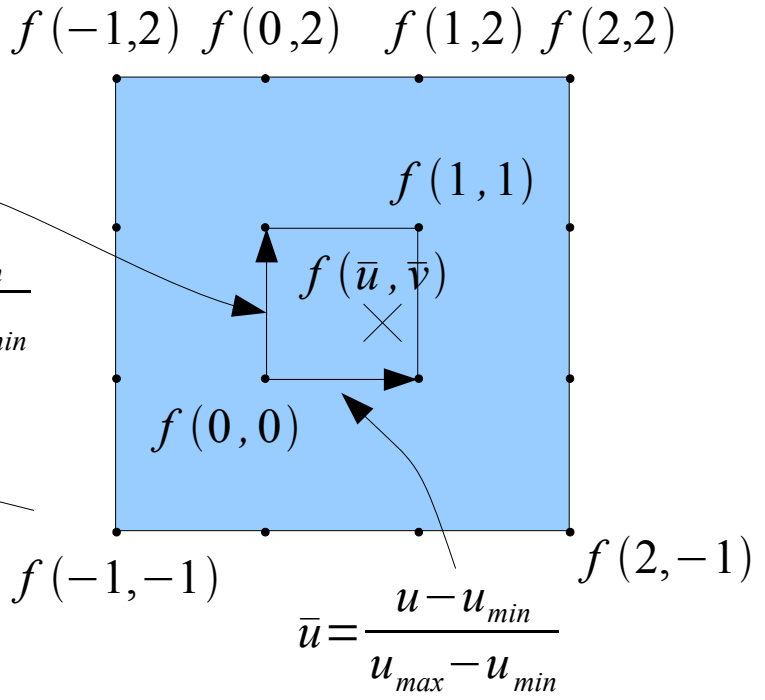
Infographie

Aliasing

- Bicubique :



$$\bar{v} = \frac{v - v_{min}}{v_{max} - v_{min}}$$



$$\bar{u} = \frac{u - u_{min}}{u_{max} - u_{min}}$$

$$\begin{aligned} f(\bar{u},\bar{v}) = {} & a_{00} + a_{01}\bar{v} + a_{02}\bar{v}^2 + a_{03}\bar{v}^3 \\ & + a_{10}\bar{u} + a_{11}\bar{u}\bar{v} + a_{12}\bar{u}\bar{v}^2 + a_{13}\bar{u}\bar{v}^3 \\ & + a_{20}\bar{u}^2 + a_{21}\bar{u}^2\bar{v} + a_{22}\bar{u}^2\bar{v}^2 + a_{23}\bar{u}^2\bar{v}^3 \\ & + a_{30}\bar{u}^3 + a_{31}\bar{u}^3\bar{v} + a_{32}\bar{u}^3\bar{v}^2 + a_{33}\bar{u}^3\bar{v}^3 \end{aligned}$$

Infographie

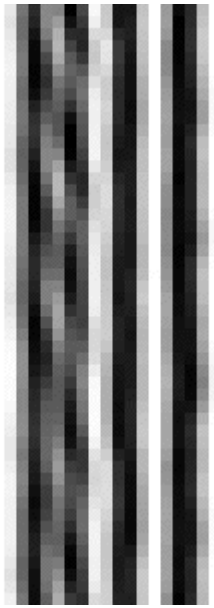
Aliasing

Ici on pose $p[i][j]=f(i\text{-}1, j\text{-}1)$

```
double bicubicInterpolate (double[][] p, double u, double v)
{
double a00 = p[1][1];
double a01 = p[1][2] - p[1][1]/2 - p[1][0]/3 - p[1][3]/6;
double a02 = p[1][0]/2 - p[1][1] + p[1][2]/2;
double a03 = p[1][1]/2 - p[1][0]/6 - p[1][2]/2 + p[1][3]/6;
double a10 = p[2][1] - p[1][1]/2 - p[0][1]/3 - p[3][1]/6;
double a11 = p[0][0]/9 + p[0][1]/6 - p[0][2]/3 + p[0][3]/18 + p[1][0]/6 + p[1][1]/4 - p[1][2]/2 + p[1][3]/12 -
p[2][0]/3 - p[2][1]/2 + p[2][2] - p[2][3]/6 + p[3][0]/18 + p[3][1]/12 - p[3][2]/6 + p[3][3]/36;
double a12 = p[0][1]/3 - p[0][0]/6 - p[0][2]/6 - p[1][0]/4 + p[1][1]/2 - p[1][2]/4 + p[2][0]/2 - p[2][1] + p[2]
[2]/2 - p[3][0]/12 + p[3][1]/6 - p[3][2]/12;
double a13 = p[0][0]/18 - p[0][1]/6 + p[0][2]/6 - p[0][3]/18 + p[1][0]/12 - p[1][1]/4 + p[1][2]/4 - p[1][3]/12 -
p[2][0]/6 + p[2][1]/2 - p[2][2]/2 + p[2][3]/6 + p[3][0]/36 - p[3][1]/12 + p[3][2]/12 - p[3][3]/36;
double a20 = p[0][1]/2 - p[1][1] + p[2][1]/2;
double a21 = p[0][2]/2 - p[0][1]/4 - p[0][0]/6 - p[0][3]/12 + p[1][0]/3 + p[1][1]/2 - p[1][2] + p[1][3]/6 - p[2]
[0]/6 - p[2][1]/4 + p[2][2]/2 - p[2][3]/12;
double a22 = p[0][0]/4 - p[0][1]/2 + p[0][2]/4 - p[1][0]/2 + p[1][1] - p[1][2]/2 + p[2][0]/4 - p[2][1]/2 + p[2]
[2]/4;
double a23 = p[0][1]/4 - p[0][0]/12 - p[0][2]/4 + p[0][3]/12 + p[1][0]/6 - p[1][1]/2 + p[1][2]/2 - p[1][3]/6 -
p[2][0]/12 + p[2][1]/4 - p[2][2]/4 + p[2][3]/12;
double a30 = p[1][1]/2 - p[0][1]/6 - p[2][1]/2 + p[3][1]/6;
double a31 = p[0][0]/18 + p[0][1]/12 - p[0][2]/6 + p[0][3]/36 - p[1][0]/6 - p[1][1]/4 + p[1][2]/2 - p[1][3]/12 +
p[2][0]/6 + p[2][1]/4 - p[2][2]/2 + p[2][3]/12 - p[3][0]/18 - p[3][1]/12 + p[3][2]/6 - p[3][3]/36;
double a32 = p[0][1]/6 - p[0][0]/12 - p[0][2]/12 + p[1][0]/4 - p[1][1]/2 + p[1][2]/4 - p[2][0]/4 + p[2][1]/2 -
p[2][2]/4 + p[3][0]/12 - p[3][1]/6 + p[3][2]/12;
double a33 = p[0][0]/36 - p[0][1]/12 + p[0][2]/12 - p[0][3]/36 - p[1][0]/12 + p[1][1]/4 - p[1][2]/4 + p[1][3]/12
+ p[2][0]/12 - p[2][1]/4 + p[2][2]/4 - p[2][3]/12 - p[3][0]/36 + p[3][1]/12 - p[3][2]/12 + p[3][3]/36;

double u2 = u * u; double u3 = u2 * u; double v2 = v * v; double v3 = v2 * v;
return a00      + a01 * v      + a02 * v2      + a03 * v3 +
       a10 * u  + a11 * u * v  + a12 * u * v2  + a13 * u * v3 +
       a20 * u2 + a21 * u2 * v + a22 * u2 * v2 + a23 * u2 * v3 +
       a30 * u3 + a31 * u3 * v + a32 * u3 * v2 + a33 * u3 * v3;
}
```

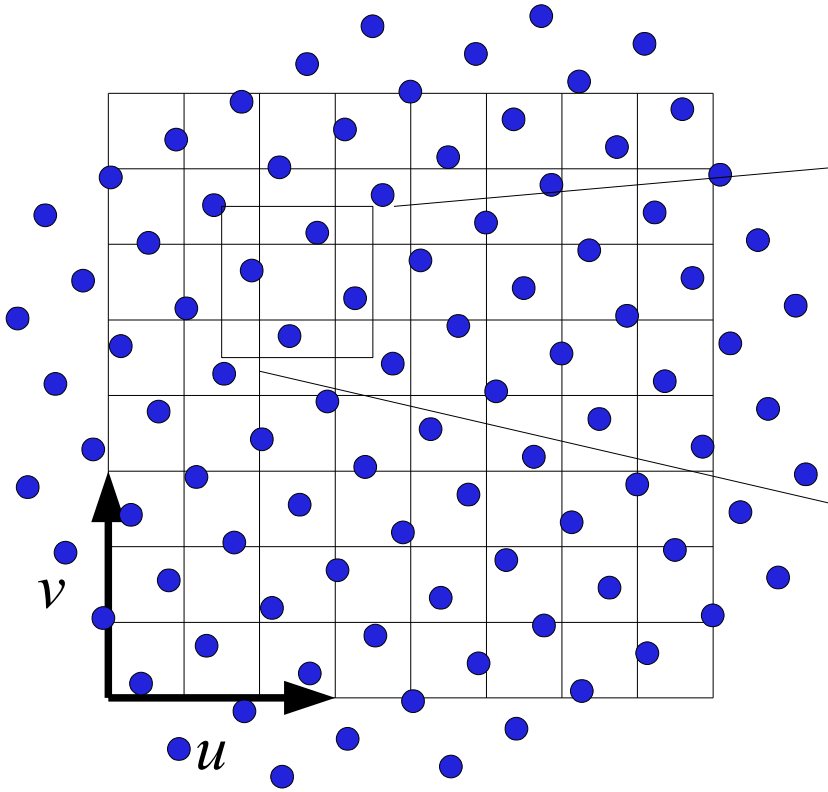
- Filtre « bicubique »

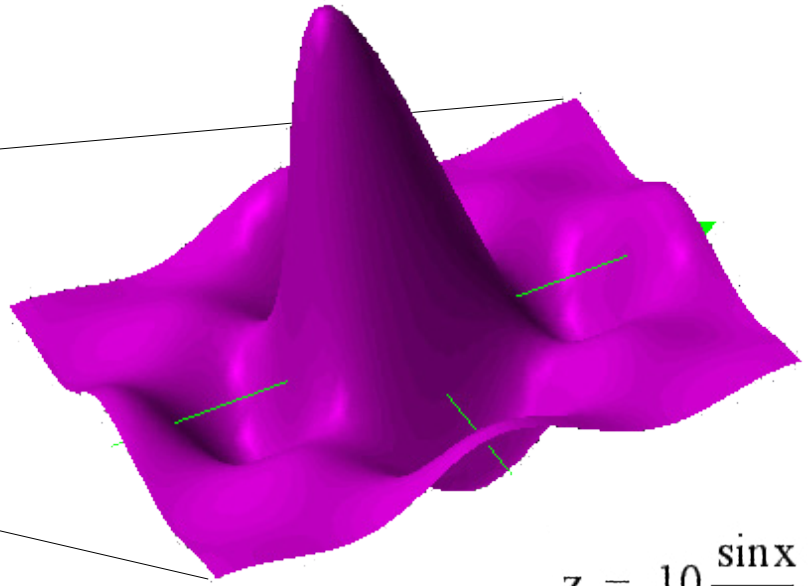


Infographie

Aliasing

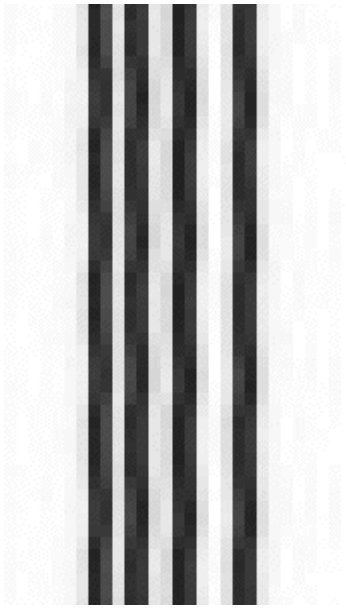
- Lanczos :





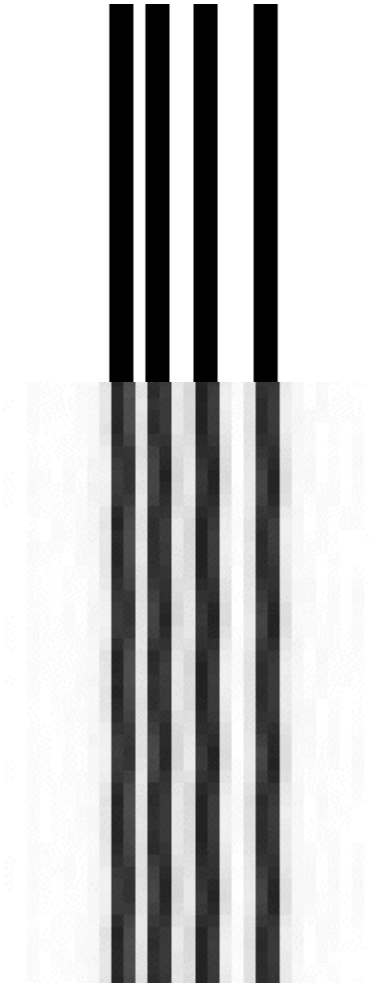
$$z = 10 \frac{\sin x}{x} \frac{\sin y}{y}$$

- Filtre « lanczos »



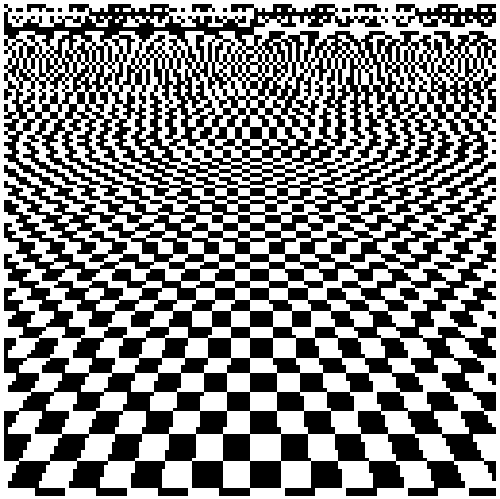
Infographie

Aliasing

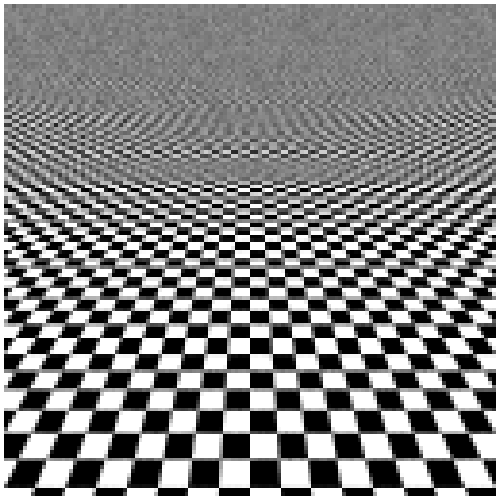


Aliasing

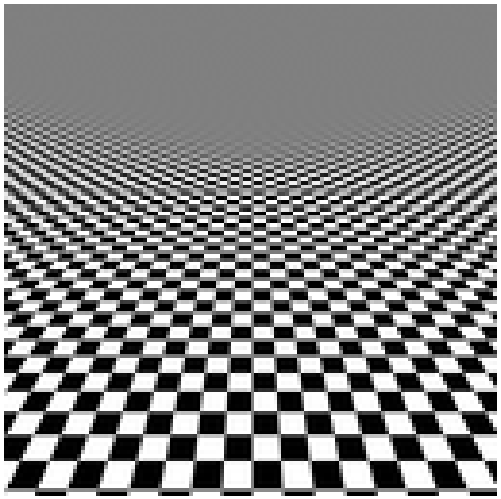
- Autre exemple



Sans antialiasing
plus proche voisin



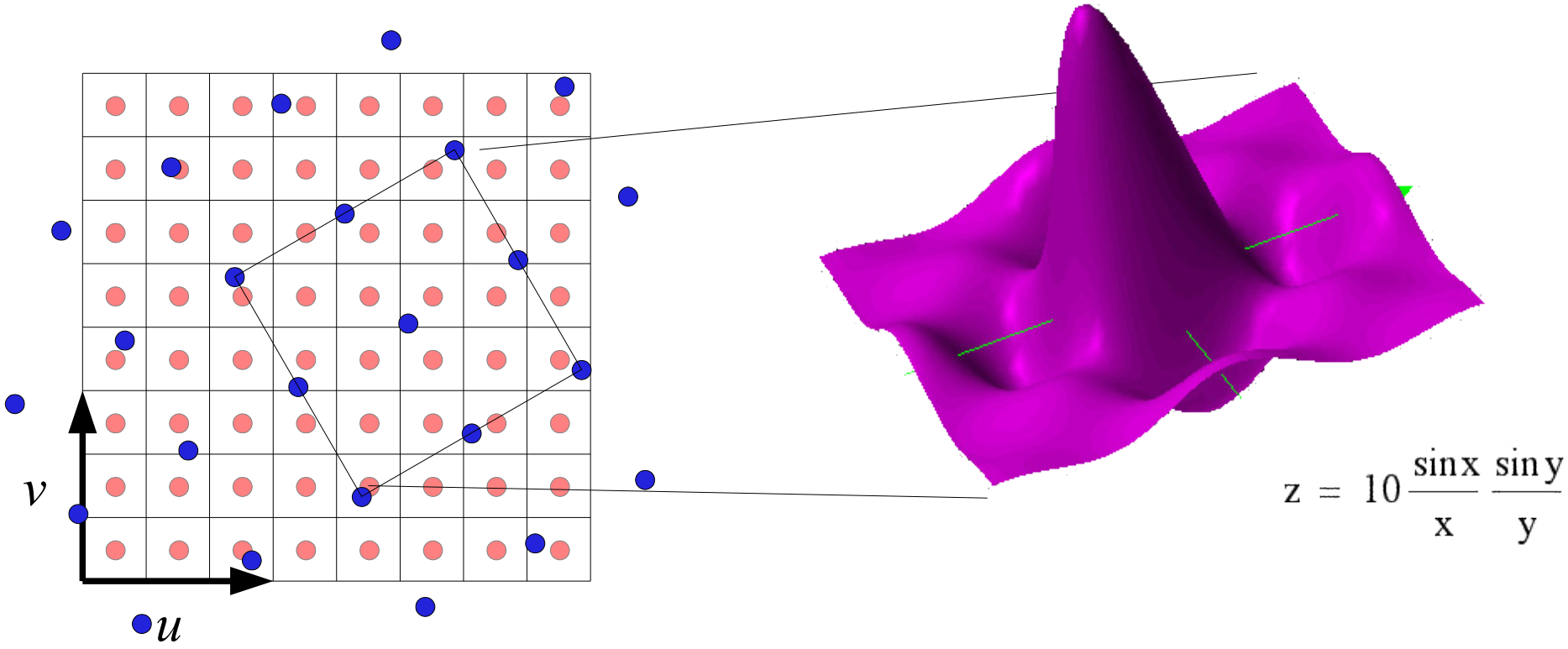
Antialiasing
par oversampling 4
et simple moyenne



Antialiasing par
filtrage Lanczos
(idéal mais non
réaliste en temps
calcul)

Aliasing

- Cas du downsampling



Il faut filtrer les données d'entrée avec un filtre passe bas défini par la résolution du pas de la grille d'arrivée....

Stockage des images

Infographie

Stockage des images

- Idée : les images ne sont pas aléatoires
  - On peut tirer parti de la structure pour stocker les images
  - Deux approches
    - Images vectorielles
    - Images discétisées
      - Compression sans pertes
      - Compression avec pertes contrôlées

tiff gif png

jpg svg cin

exr

bmp

Infographie

Stockage des images

- TIFF : format universel mais parfois implémenté partiellement
- JPG : limité à 8bits/canal, compression DCT avec pertes (WWW)
- PNG : format ouvert, 1/2/4/8 bits indexé, 8/16 bits/ canal; canal alpha (transparence), compression type LZW – pas de brevet (WWW)
- GIF : indexé 8 bits, transparence (1 bit), compression type LZW, animation possible, brevet échu (WWW)
- SVG : images vectorielles
- CIN : vieux format « cineon » 10 bits / canal utilisé pour effets spéciaux. Compression sans pertes
- EXR : format ouvert Lucasfilm (ILM): format 16/32 bits par canal en virgule flottante, compression sans perte
- BMP : vieux format windows sans compression limité à 8 bits par canal + transparence.

Infographie

Stockage des images

- Comment choisir ?
  - Dessin au trait, destiné à être mis à l'échelle --- format vectoriel
  - Images en général, format discrétisé

Infographie

Stockage des images

- Images vectorielles
  - En général pas de compression
  - Exemple type : fontes de caractères destinées à être agrandies
  - Idéal comme format pour des dessins « au trait »

Wmf : windows metafile (exclusivement windows)

Svg : Scalable vector graphics (standard ouvert)

+ logiciels propriétaires et leurs formats : coreldraw, adobe illustrator...

Infographie

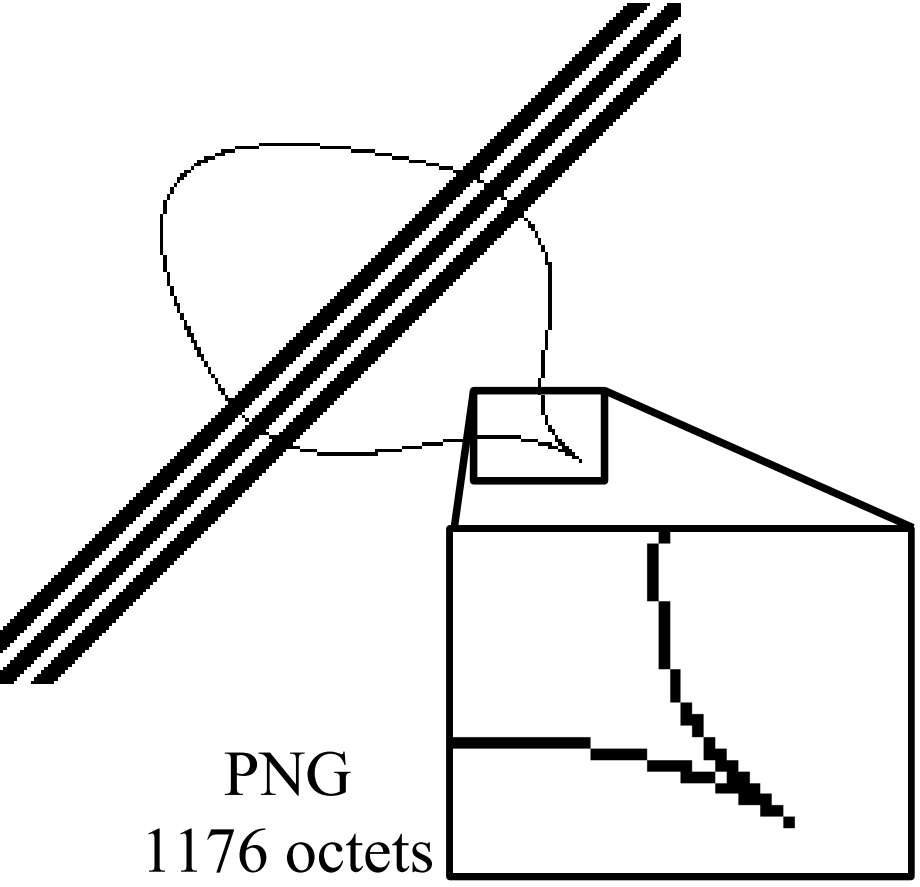
Stockage des images

- Images discrétisées (photos ou autres)
  - Stockage sans compression
    - bmp (vieux), tiff (bits, 16 bits, virg. flottante) , png (8,16 bits/c, canal alpha, coul. Indexées 1,2,4,8 bits), gif (couleurs indexées + canal alpha), exr(virgule flottante 16,32 bits/canal)
  - Compression sans pertes
    - tiff, png, gif, exr
  - Compression avec pertes contrôlées
    - tiff, jpg(8 bits), jpeg2000 (jpg amélioré mais peu utilisé)

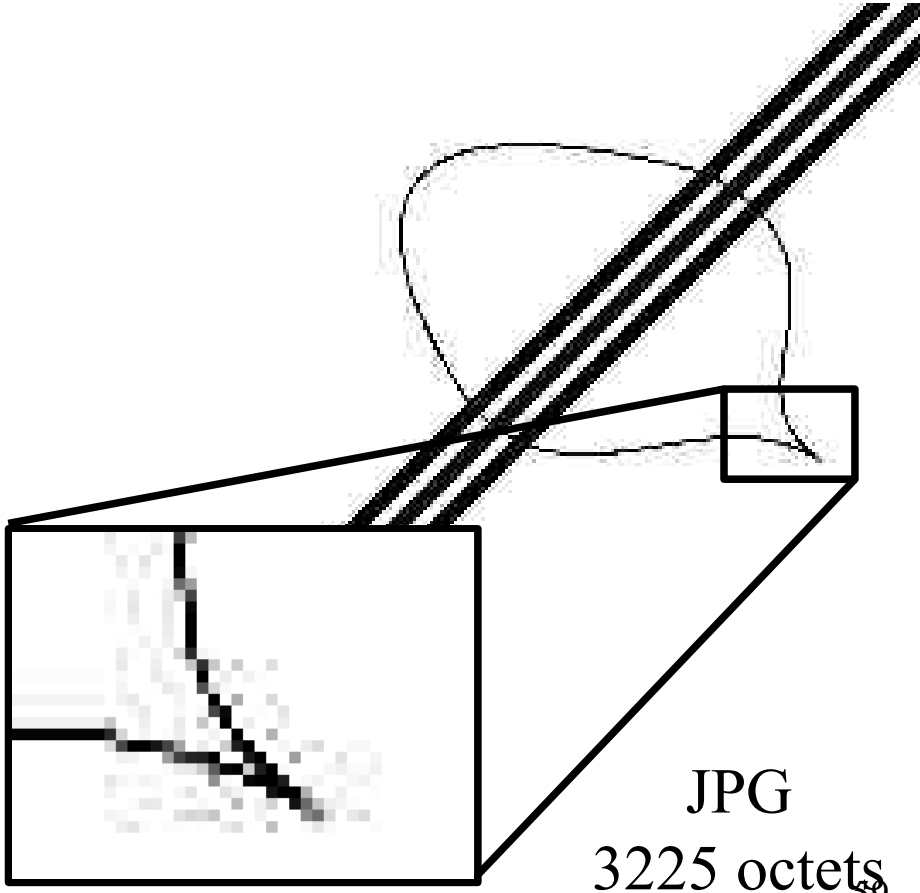
Infographie

Stockage des images

- Pour des images avec un faible nombre de couleurs, et/ou avec des traits francs, ne jamais utiliser jpg.



PNG
1176 octets



JPG
3225 octets

Stockage des images

- À l'inverse, des photographies ou des images aux tons continus sont adaptés à JPG.

JPG 24 bits: 85Ko

Infographie

Stockage des images

- À l'inverse, des photographies ou des images aux tons continus sont adaptés à JPG.

PNG 24 bits sans pertes : 646Ko

Stockage des images

- À l'inverse, des photographies ou des images aux tons continus sont adaptés à JPG.

PNG 8bits (256 couleurs) : 227 Ko

Stockage des images

- À l'inverse, des photographies ou des images aux tons continus sont adaptés à JPG.

PNG 4 bits (16 couleurs) : 108 Ko

Stockage des images

- Images à fort contraste (HDR) ou images destinées à être manipulées
  - PNG 16 bits
  - TIFF 16 bits
  - EXR
- Images destinées au WWW ou à l'affichage sur écran de bureau
  - GIF, PNG, JPG
- Si la place n'est pas un problème, toujours privilégier une compression sans pertes et une profondeur (nb bits/plans) élevée.

Stockage des images

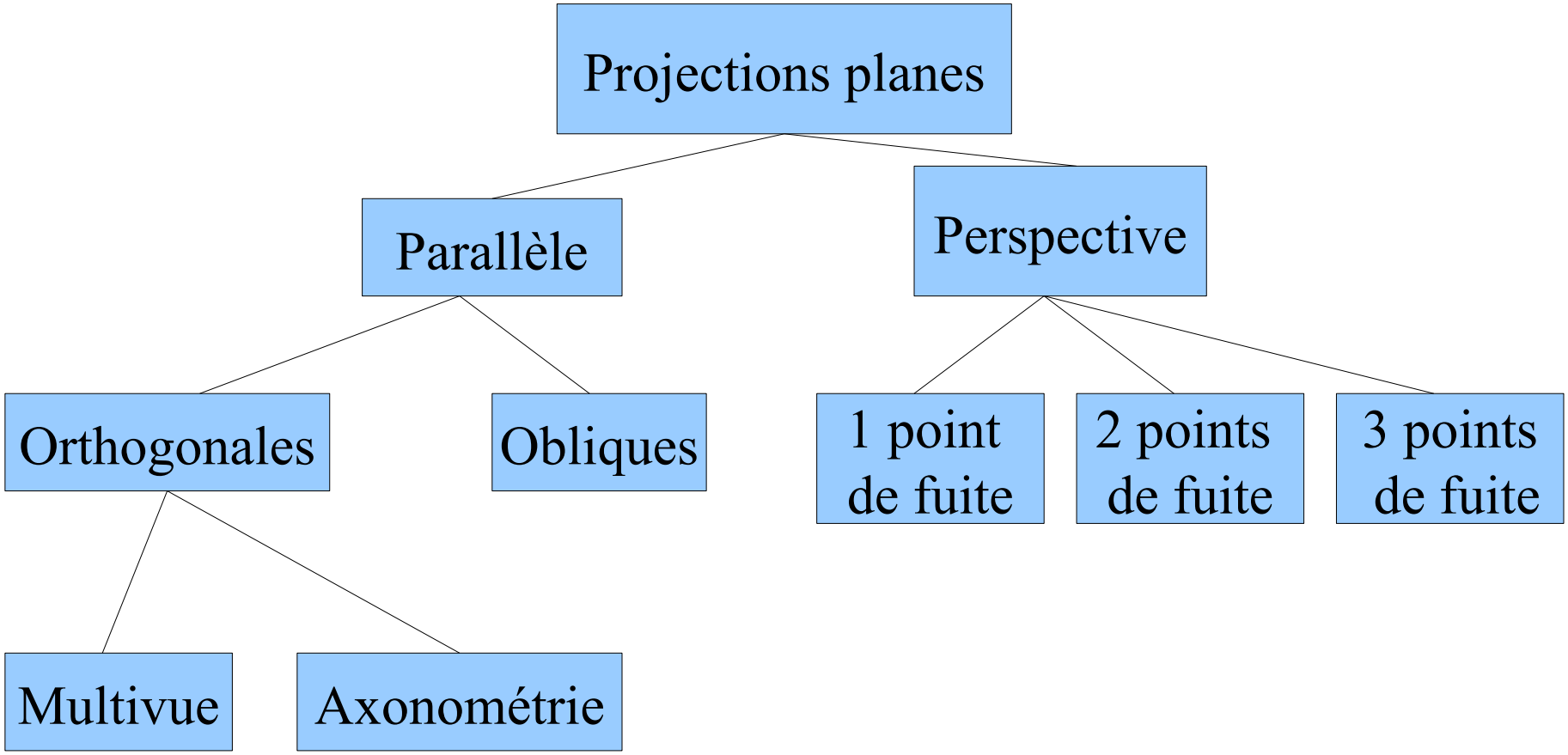
Il n'existe pas de format d'image universel !

Perspective et matrices de transformations

Infographie

Perspective

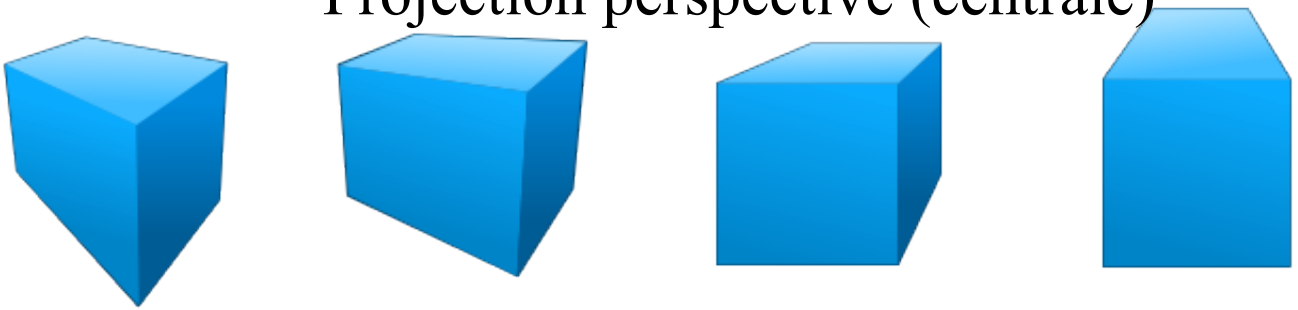
- Projections « classiques »



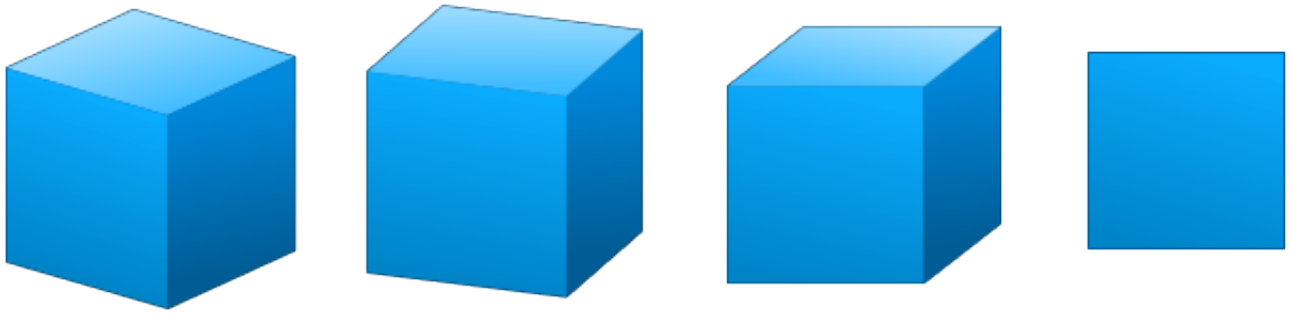
Infographie

Perspective

Projection perspective (centrale)



Projection parallèle



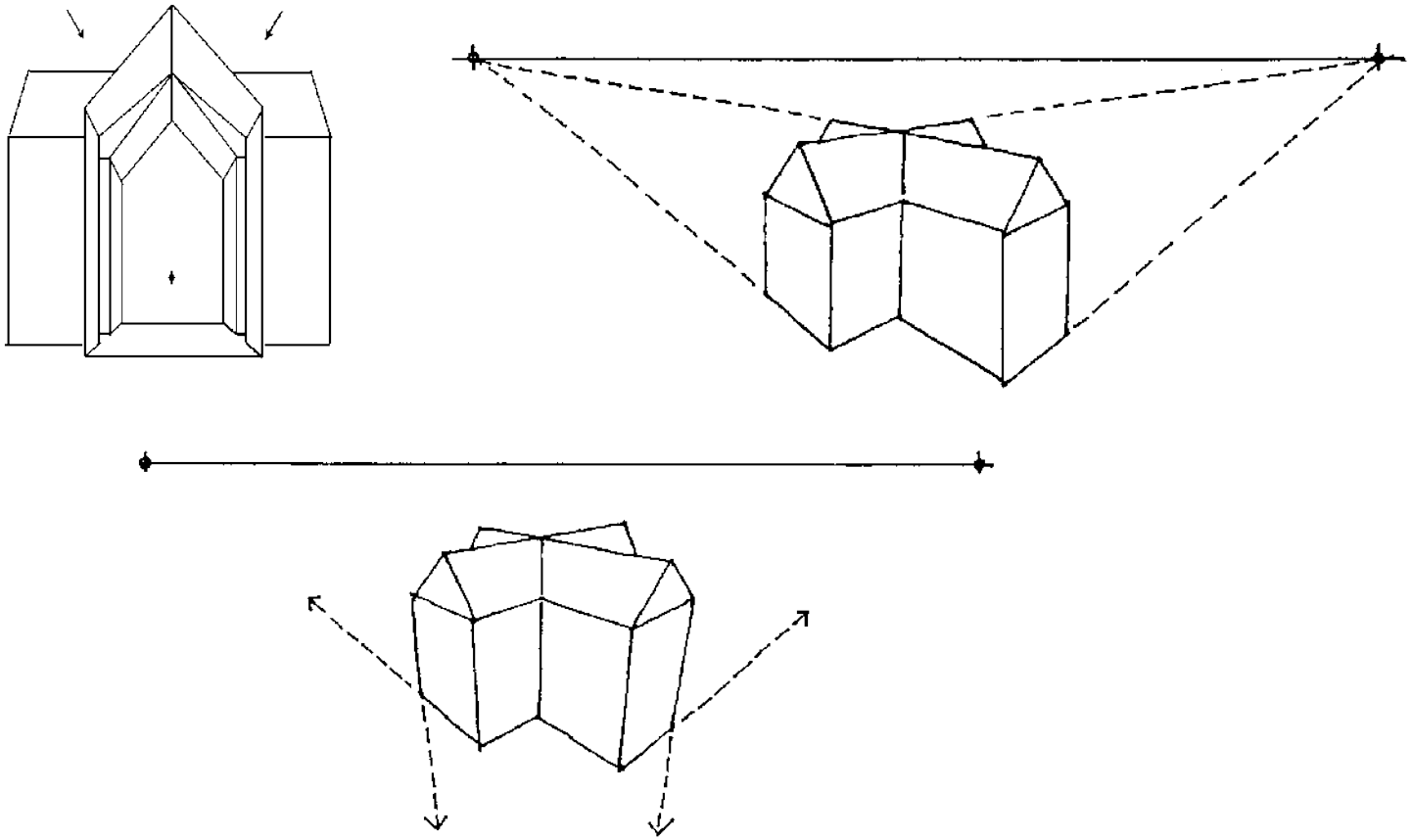
Projection oblique

Axonométrie
Orthogonale

Infographie

Perspective

- Points de fuite en projection perspective



Université de Liège

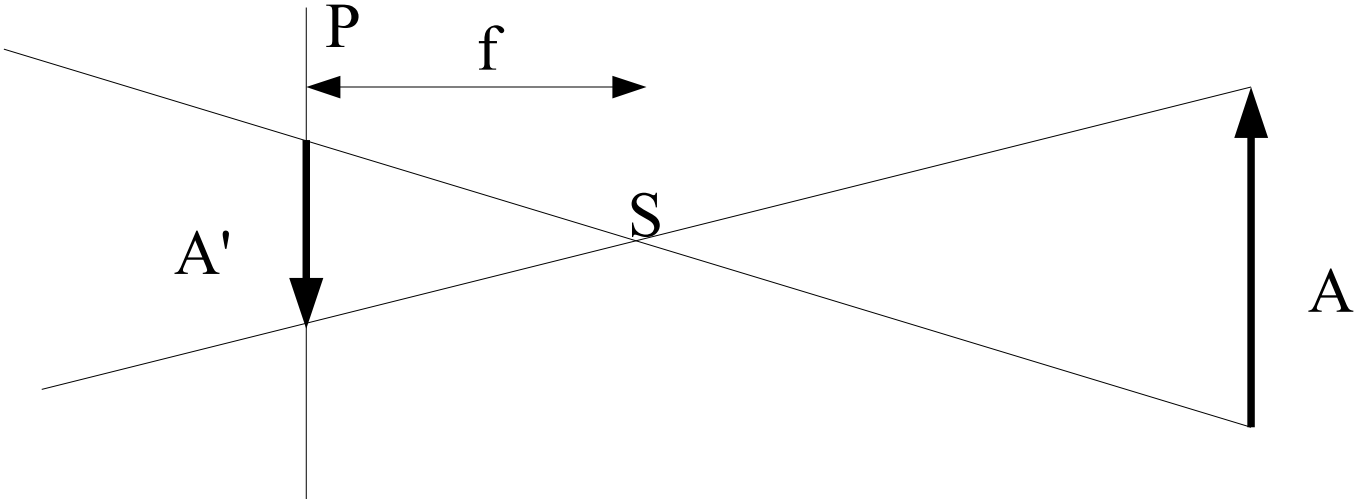
Infographie

Perspective

Infographie

Perspective

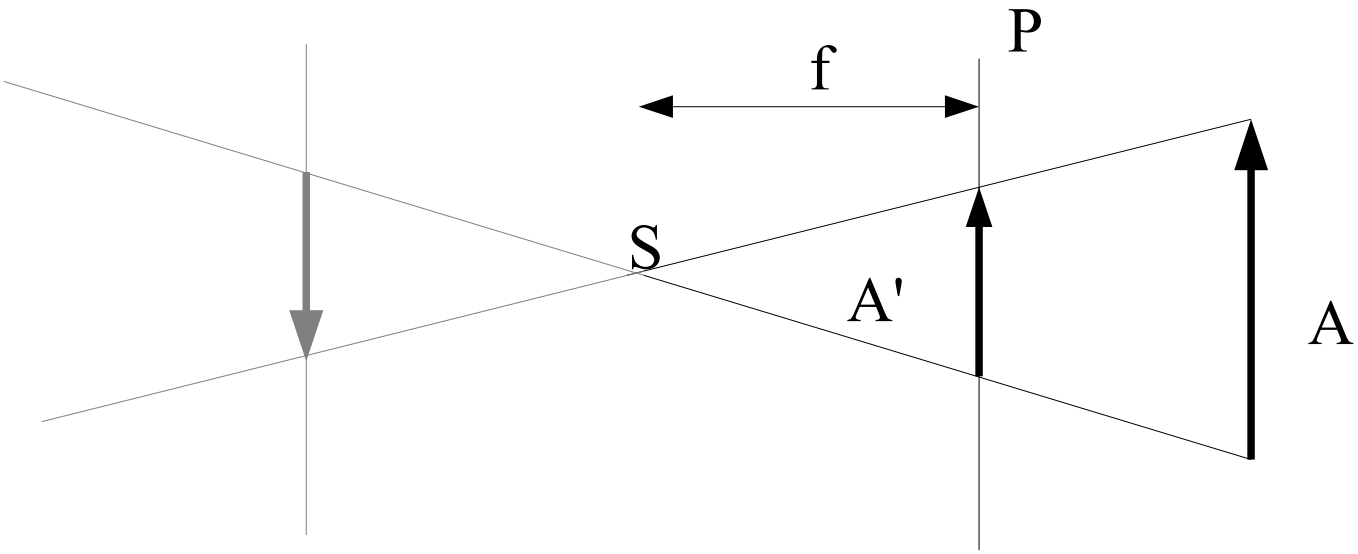
- La perspective est une représentation assez exacte de ce que l'œil voit
  - Basée sur la projection centrale
  - En première approximation, l'œil (ou un appareil photo) est constitué d'un « objectif » se réduisant à un point (le cristallin), et d'un plan de projection de l'image (rétine)



Infographie

Perspective

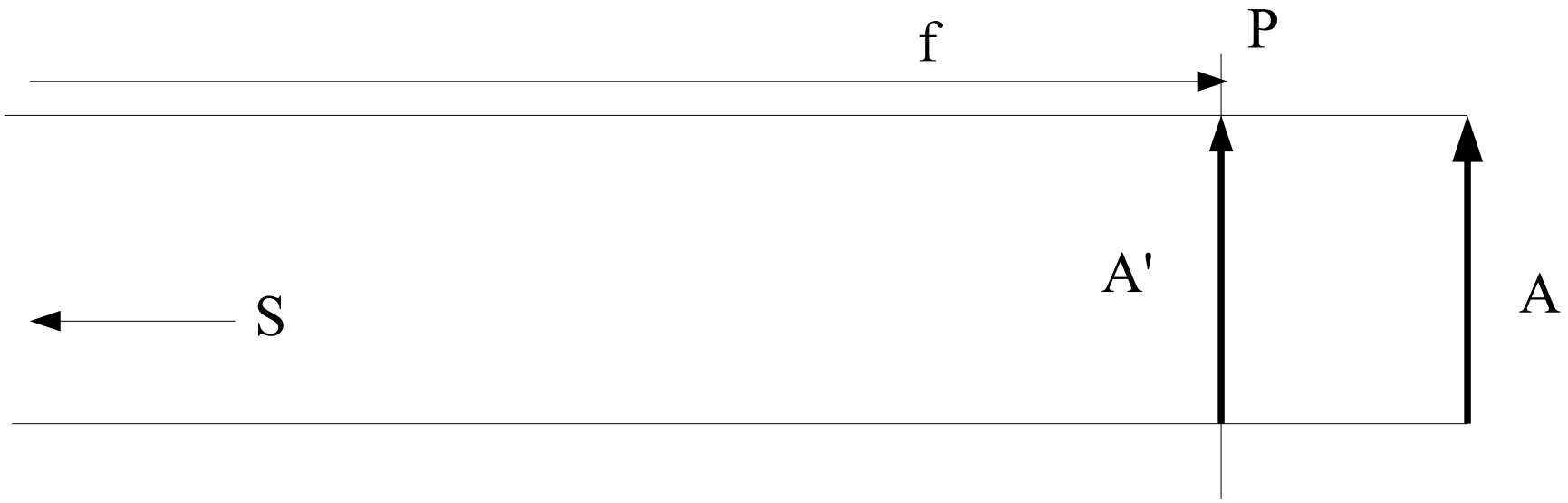
- Configuration équivalente utilisée en infographie



Infographie

Perspective

- La projection parallèle correspond au cas limite ou f tend vers l'infini.



Transformations géométriques

Matrices de transformation

- Transformations géométriques
  - Deux buts
    - Obtenir à partir des objets aux coordonées 3D une projection sur le plan de l'ecran (coordonnées 2D + profondeur)
    - A partir d'objets élémentaires, pouvoir les placer n'importe ou dans le volume, éventuellement affublés d'opérations telles que le cisaillement ou la mise à l'échelle.

Transformations affines

- Cas des transformation linéaires
  - Transformations affines

$$\phi(\boldsymbol{P}) \equiv \mathbf{A} \cdot \boldsymbol{P} + \boldsymbol{u}, \quad \boldsymbol{u} \in \mathbb{R}^3$$

Infographie

Quelques transformations affines

identité : $\boldsymbol{u} = 0$, $\mathbf{A} = \mathbf{I}$, $\mathbf{I}$ est la matrice identité,

$$\boldsymbol{u} = 0 \;;\; \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

translation :
$\boldsymbol{u}$ est le vecteur de translation, $\mathbf{A} = \mathbf{I}$,

$$\boldsymbol{u} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \;;\; \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

mise à échelle
$\boldsymbol{u} = 0$, $\mathbf{A}$ est une matrice diagonale dont les termes définissent les échelles selon les axes,

$$\boldsymbol{u} = 0 \;;\; \mathbf{A} = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix}$$

rotation :
$\boldsymbol{u} = 0$, $\mathbf{A}$ est une matrice de rotation,

$$\boldsymbol{u} = 0 \;;\; \mathbf{A} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Quelques transformations affines

cisaillement :
où a, b, c sont les 3 coefficients de cisaillement.

$$\boldsymbol{u} = 0 \;\; ; \;\; \mathbf{A} = \begin{bmatrix} 1 & a & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{bmatrix}$$

Cas particulier important :
Si la matrice **A** est orthogonale : $\mathbf{A}^T = \mathbf{A}^{-1}$

alors cette transformation conserve les angles et les longueurs.

Traitement matriciel

Traitement multiplicatif

$$P_1 = \mathbf{S} \cdot P_0$$

$$P_1 = \mathbf{R} \cdot P_0$$

$$P_1 = \mathbf{C} \cdot P_0$$

Traitement additif de la translation

$$P_1 = P_0 + t$$

Le traitement n'est pas le même pour toutes les opérations...

Infographie

Coordonnées homogène

- De façon à rendre le traitement identique de la translation, on rajoute une coordonnée en plus, fixée pour le moment à 1 :
- La coordonnée supplémentaire servira pour la projection perspective dans la suite

$$\begin{bmatrix} 1 & 0 & 0 & r \\ 0 & 1 & 0 & s \\ 0 & 0 & 1 & t \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x+r \\ y+s \\ z+t \\ 1 \end{bmatrix}$$

Autres opérations

Translation

Infographie

Matrices de transformation

- translation : $\mathbf{D}(\boldsymbol{t}) = \begin{bmatrix} 1 & 0 & 0 & u \\ 0 & 1 & 0 & v \\ 0 & 0 & 1 & w \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ; \quad \boldsymbol{t} = \begin{bmatrix} u \\ v \\ w \\ 1 \end{bmatrix}$

- mise à échelle : $\mathbf{S}(p, q, r) = \begin{bmatrix} p & 0 & 0 & 0 \\ 0 & q & 0 & 0 \\ 0 & 0 & r & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Infographie

Matrices de transformation

- rotation autour de l'axe z : $$\mathbf{R}_z(\theta)=\begin{bmatrix}\cos\theta & -\sin\theta & 0 & 0\\ \sin\theta & \cos\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

- rotation autour de l'axe x : $$\mathbf{R}_x(\theta)=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\theta & -\sin\theta & 0\\ 0 & \sin\theta & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

- rotation autour de l'axe y : $$\mathbf{R}_y(\theta)=\begin{bmatrix}\cos\theta & 0 & \sin\theta & 0\\ 0 & 1 & 0 & 0\\ -\sin\theta & 0 & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Infographie

Matrices de transformation

- cisaillement (cas général) $\mathbf{C}(a,b,c)=\begin{bmatrix} 1 & a & b & 0 \\ 0 & 1 & c & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

- On peut combiner ces matrices de transformation par simple multiplication
  - Respecter l'ordre (pas commutatif !)

$$\mathbf{G}_n=\mathbf{G}_{n-1}\cdots\mathbf{G}_2\cdot\mathbf{G}_1$$

Matrices de transformation

- En particulier, on exprime une transformation supplémentaire T:
  - Par rapport à l'origine du repère (O)

$$\mathbf{G}_n = \mathbf{T} \cdot \mathbf{G}_{n-1}$$

  - Par rapport à la référence d'un objet (transformée de O par $\mathbf{G}_{n-1}$)

$$\mathbf{G}_n = \mathbf{G}_{n-1} \cdot \mathbf{T}$$

  - Par rapport à un point A quelconque

$$\mathbf{G}_n = \mathbf{D}(x_A, y_A, z_A) \cdot \mathbf{T} \cdot \mathbf{D}(-x_A, -y_A, -z_A) \cdot \mathbf{G}_{n-1}$$

Infographie

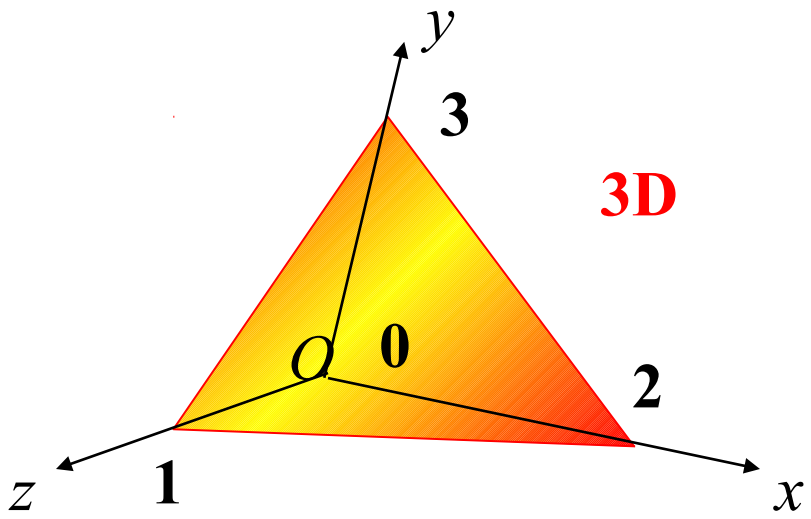
Matrices de transformation

- Changement de repère (pour axonométries ou projection parallèle)
  - Repère 3D global vers repère écran 2D
  - On a 12 paramètres dans la transformation
  - On peut donc spécifier la transformation de façon univoque par 4 points (formant un tétraèdre) et leur transformée
    - Application du théorème de Pohlke

Infographie

Matrices de transformation

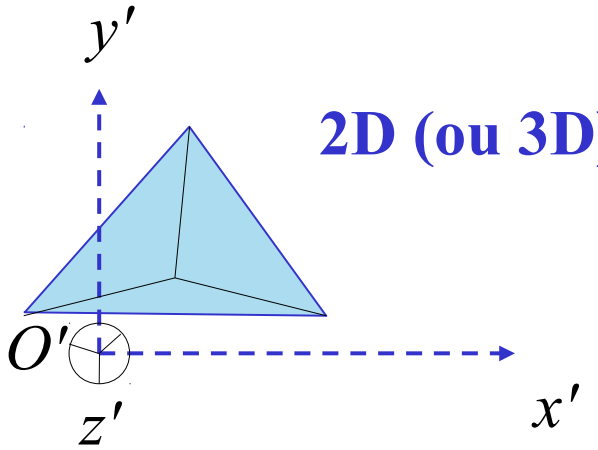
$$P_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$



$$P_1 = \begin{bmatrix} x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \\ z_0 & z_1 & z_2 & z_3 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$P_1 = \mathbf{T} \cdot P_0$$

$$\mathbf{T} = P_1 \cdot P_0^{-1}$$



Infographie

Matrices de transformation

$$P_0 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \quad P_0^{-1} = \begin{bmatrix} -1 & -1 & -1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad P_1 = \begin{bmatrix} x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \\ z_0 & z_1 & z_2 & z_3 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{T} = \begin{bmatrix} x_1 - x_0 & x_2 - x_0 & x_3 - x_0 & x_0 \\ y_1 - y_0 & y_2 - y_0 & y_3 - y_0 & y_0 \\ z_1 - z_0 & z_2 - z_0 & z_3 - z_0 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Cette matrice permet de projeter les coordonnées dans un système d'axes quelconques

Matrices de transformation

- A l'aide des transformations que l'on vient de voir , on peut :

  - Passer des coordonnées de l'espace (x,y,z) aux coordonées d'un écran (x',y',z'=profondeur) en exprimant la position dans l'espace du point de vue lié à l'écran

  - Tenir compte d'un facteur d'échelle pour ajuster la dimension virtuelle de l'écran dans l'espace (x,y,z) et passer de (x,y) à (i,j) qui est la position géométrique sur l'écran

- La troisième coordonnée (profondeur z) peut servir pour le calcul des faces cachées.

- Cas des points pour lesquels la 4ème coordonnée est différente de 1
  - On considère que les points situés le long d'une ligne à partir de l'origine sont tous équivalents
  - Cela correspond à une projection centrale sur l'hyperplan $w=1$; les deux points suivants sont équivalents :

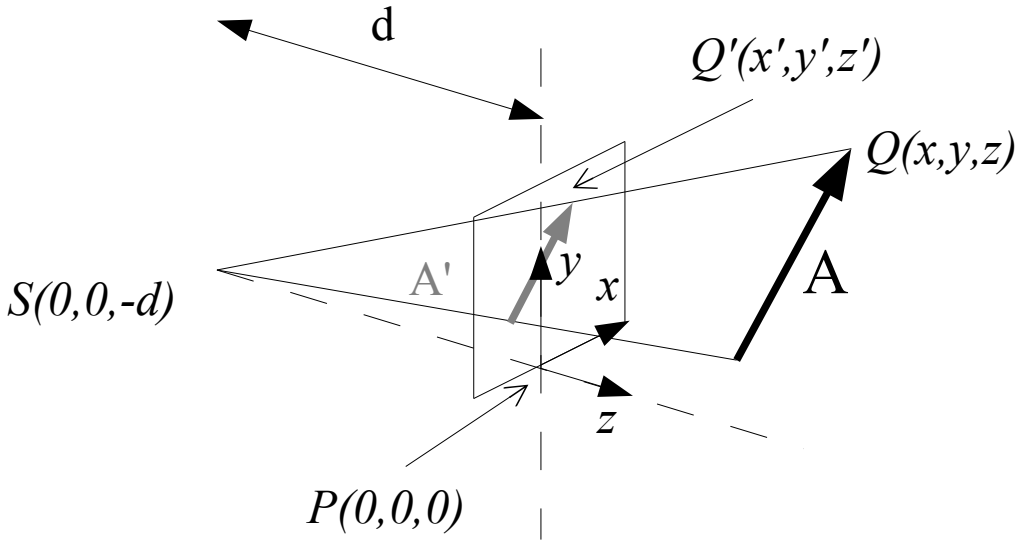
$$[wx, wy, wz, w] \Leftrightarrow [x, y, z, 1]$$

  - Si $w=0$, on parle de *vecteur* en coordonnées homogènes

$$[x, y, z, 0]$$

Matrices de transformation

- Transformation perspective
  - On va se placer dans un cas particulier qui n'ôte rien à la généralité de l'approche
    - P est l'origine O du repère
    - L'écran est un plan de normale (0,0,1) (perpendiculaire à z)



$$\frac{x'}{x} = \frac{y'}{y} = \frac{d}{(d+z)}$$

$$z' = 0$$

Matrices de transformation

- Coordonnées écran en fonction des coordonnées espace

$$x' = \frac{x}{1+\frac{z}{d}} \qquad y' = \frac{y}{1+\frac{z}{d}} \qquad z' = 0$$

  - On va s'arranger pour que la 3ème coordonnée subisse la même mise à l'échelle

$$x' = \frac{x}{1+\frac{z}{d}} \qquad y' = \frac{y}{1+\frac{z}{d}} \qquad z' = \frac{z}{1+\frac{z}{d}}$$

Matrices de transformation

- Les trois coordonnées cartésiennes :

$$x' = \frac{x}{1+\frac{z}{d}} \qquad y' = \frac{y}{1+\frac{z}{d}} \qquad z' = \frac{z}{1+\frac{z}{d}}$$

peuvent être exprimées sous forme de coordonnées homogènes

$$\left(\frac{x}{1+\frac{z}{d}}, \frac{y}{1+\frac{z}{d}}, \frac{z}{1+\frac{z}{d}}, 1 \right) \equiv (x, y, z, 1+\frac{z}{d})$$

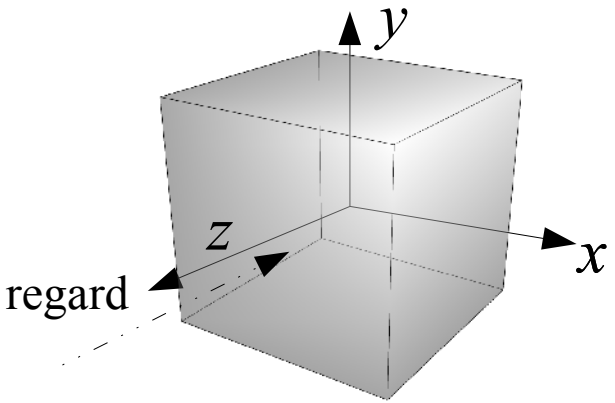
Matrices de transformation

- On peut donc écrire la transformation sous forme de transformation linéaire (un cisaillement) dans l'espace homogène

$$\begin{pmatrix} x \\ y \\ z \\ 1+\frac{z}{d} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{d} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

  - Les opération suivantes conduisent à l'image désirée :
    - Le cisaillement en 4D
    - Puis une projection perspective vers l'espace 3D (par la division perspective)
    - Puis une projection orthogonale sur l'écran z=cste

- Champ de vision et construction effective des matrices de transformation
  - Champ de vision canonique
    - $(x_c, y_c, z_c) \in [-1,1]^3$
    - Ecran : constitué par (n_x, n_y) pixels, centré à l'origine, dans le plan (x_c, y_c) : $(x_p, y_p) \in [-0.5, n_x - 0.5] \times [-0.5, n_y - 0.5]$
    - Transformation du champ canonique vers les coordonnées écran



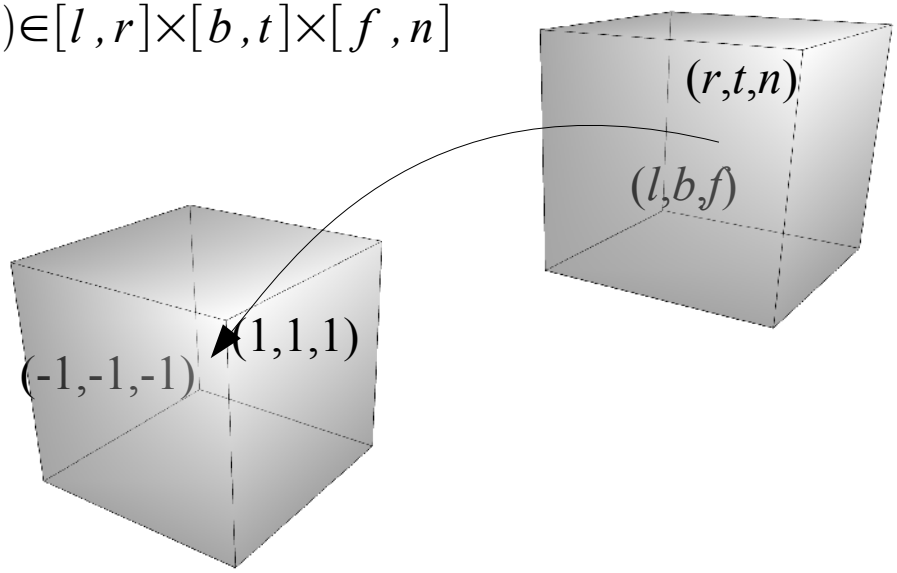
$$\begin{pmatrix} x_p \\ y_p \\ z_p \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x - 1}{2} \\ 0 & -\frac{n_y}{2} & 0 & \frac{n_y - 1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_s} \cdot \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}$$

NB : Si y_c est inversé, il faut en tenir compte ...

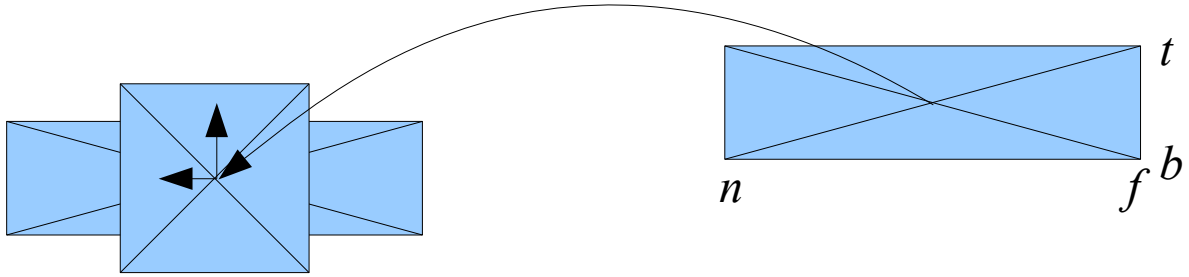
Matrices de transformation

- Projection orthographique
  - On va faire correspondre un volume déterminé avec le volume canonique
    - Ce volume est aligné avec le volume canonique mais n'a pas le même centre, ni les mêmes dimensions

$$(x_d, y_d, z_d) \in [l, r] \times [b, t] \times [f, n]$$



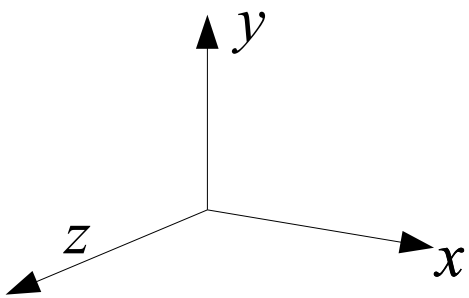
- La transformation revient à une translation suivie d'une dilatation.

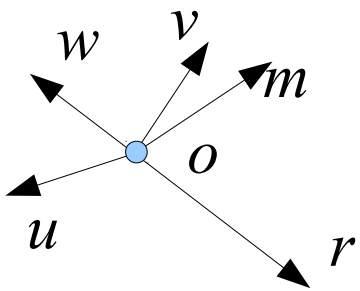


$$\begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{n-f} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -\frac{r+l}{2} \\ 0 & 1 & 0 & -\frac{b+t}{2} \\ 0 & 0 & 1 & -\frac{n+f}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_c} \cdot \begin{pmatrix} x_d \\ y_d \\ z_d \\ 1 \end{pmatrix}$$

Matrices de transformation

- Orientation arbitraire du point de vue
  - On voudrait pouvoir regarder dans une direction arbitraire et à partir d'un point quelconque
    - On définit la position de l'œil (o), la direction du regard (r) et le « midi » -verticale selon l'observateur- (m)
    - On détermine un repère orthonormé (o,u,v,w) à partir de ces données





$$w = -\frac{r}{\|r\|}$$

$$u = -\frac{m \times w}{\|m \times w\|}$$

$$v = w \times u$$

Matrices de transformation

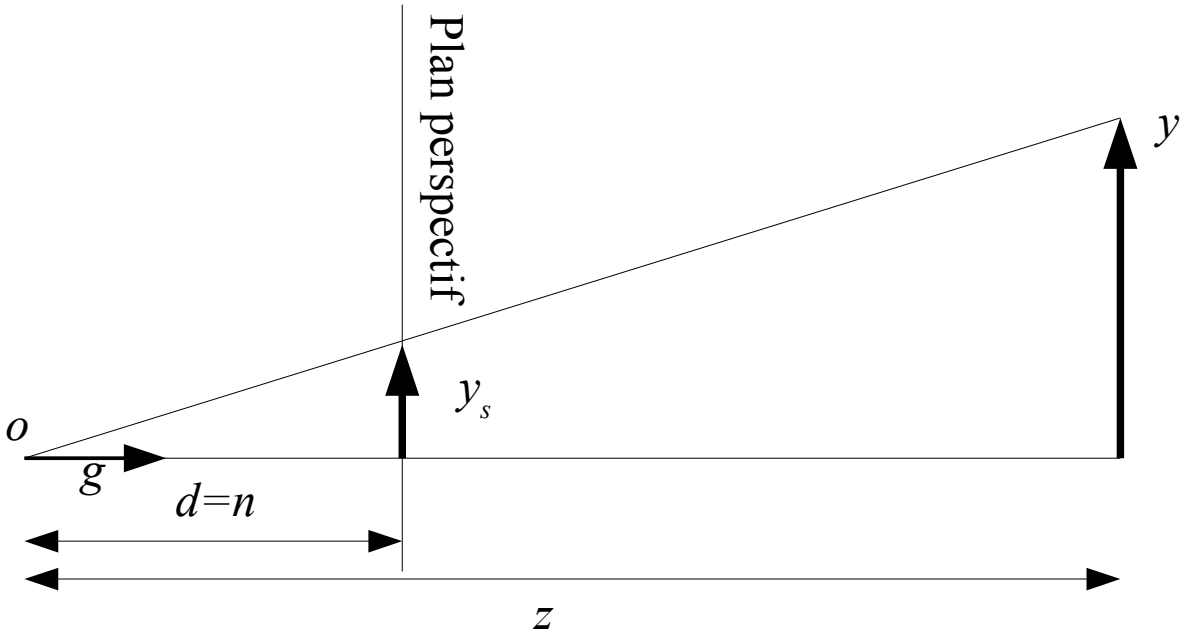
- L'alignement des coordonnées de l'espace avec celles de l'observateur se fait avec deux transformations
  - Translation amenant les coordonnées de l'œil vers l'origine
  - Rotation autour des axes pour les aligner avec les axes globaux

$$\begin{pmatrix} x_d \\ y_d \\ z_d \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} x_u & y_u & z_u & 0 \\ x_v & y_v & z_v & 0 \\ x_w & y_w & z_w & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -x_o \\ 0 & 1 & 0 & -y_o \\ 0 & 0 & 1 & -z_o \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_v} \cdot \begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix}$$

  - On peut aussi utiliser le théorème de Pohlke...à l'envers !

Matrices de transformation

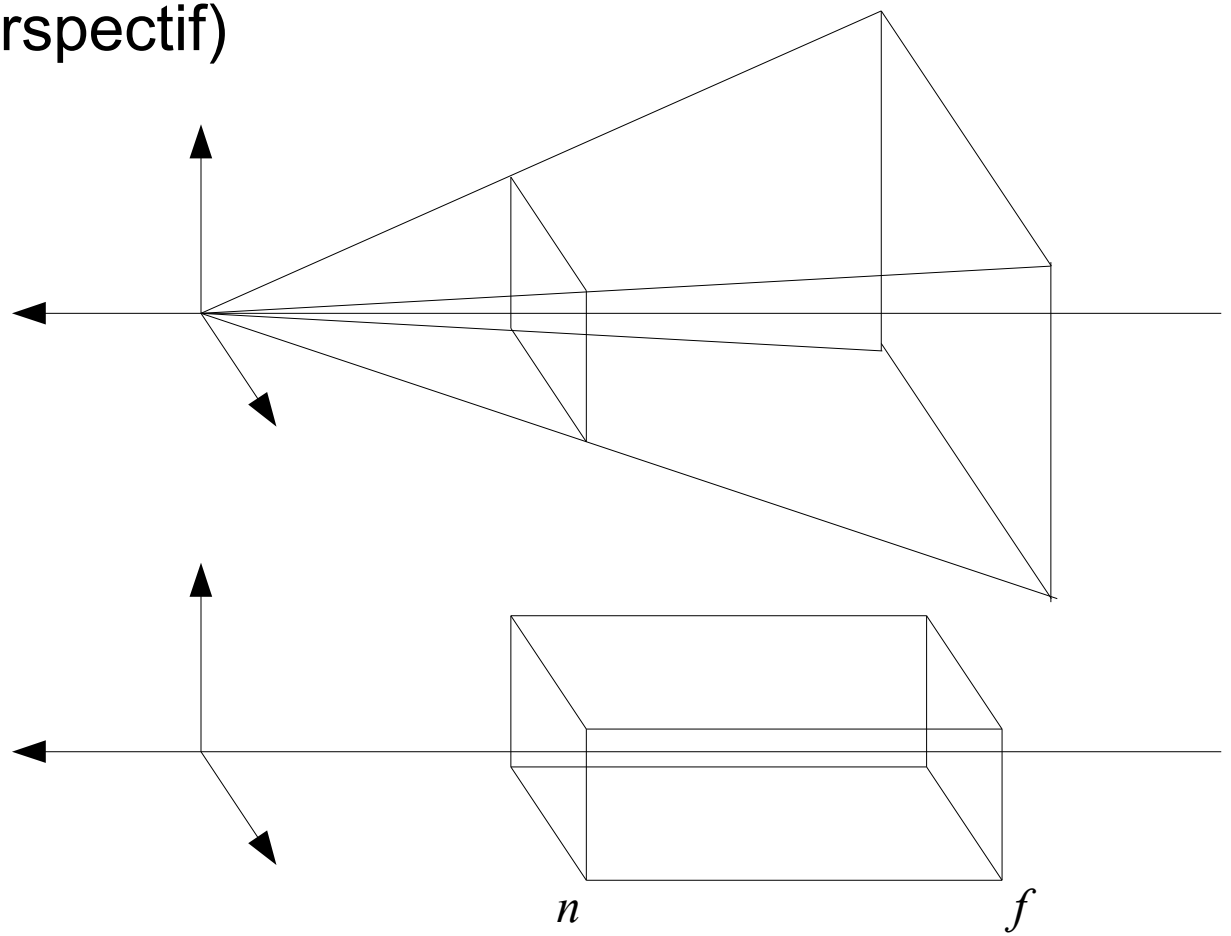
- Transformation perspective



$$y_s = \frac{d}{z} y$$

Matrices de transformation

- On voudrait ici garder la valeur « z » sur le plan f, et conserver (x,y,z) sur le plan n (plan de projection ou plan perspectif)



Infographie

Matrices de transformation

- Pour utiliser les coordonnées homogènes,
  - Les trois composantes doivent être divisées par la même valeur
  - Reprenons ce que l'on a vu :

$$\begin{pmatrix} x \\ y \\ z \\ 1+\frac{z}{d} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{d} & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

    - Par rapport à cette matrice, on a un déplacement pour amener l'oeil en (0,0,0)
    - Et une petite modification pour laisser les points en z=f inchangés

- Déplacement de $(0,0,-n)$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{n} & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -n \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -n \\ 0 & 0 & \frac{1}{n} & 0 \end{pmatrix}$$

- Garder $z = f$ et $z = n$ inchangés

$$\begin{pmatrix} x\frac{n}{z} \\ y\frac{n}{z} \\ (z\alpha+\beta)\frac{n}{z} \\ 1 \end{pmatrix} \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \alpha & \beta \\ 0 & 0 & \frac{1}{n} & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$\begin{cases} n\dfrac{(n\alpha+\beta)}{n} = n \\ n\dfrac{(f\alpha+\beta)}{f} = f \end{cases} \Rightarrow \begin{cases} \alpha = \dfrac{n+f}{n} \\ \beta = -f \end{cases}$$

Matrices de transformation

- Matrice perspective

$$\begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix} \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{n+f}{n} & -f \\ 0 & 0 & \frac{1}{n} & 0 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

  - On peut multiplier la matrice par une constante arbitraire (à cause des coordonnées homogènes)

$$\begin{pmatrix} x_e \\ y_e \\ z_e \\ 1 \end{pmatrix} \equiv \underbrace{\begin{pmatrix} n & 0 & 0 & 0 \\ 0 & n & 0 & 0 \\ 0 & 0 & n+f & -f\,n \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{M_p} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Matrices de transformation

- La matrice perspective que nous venons de définir suppose que l'on regarde dans le sens des z négatifs
  - On doit donc l'appliquer *après* le changement de point de vue !
  - La chaîne complète de transformations est donc :

$$M = M_s \cdot M_c \cdot M_p \cdot M_v$$

Infographie

Matrices de transformation

- En particulier, la matrice

$$M_{proj\_persp} = M_c \cdot M_p = \begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{l+r}{l-r} & 0 \\ 0 & \frac{2n}{t-b} & \frac{b+t}{b-t} & 0 \\ 0 & 0 & \frac{f+n}{n-f} & \frac{2fn}{f-n} \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

est appelée matrice de projection perspective et permet de passer de l'espace réel vers le volume canonique $[-1,1]^3$

$$M = M_s \cdot M_{proj\_persp} \cdot M_v$$

$$z = -|n| \rightarrow z_c = 1$$

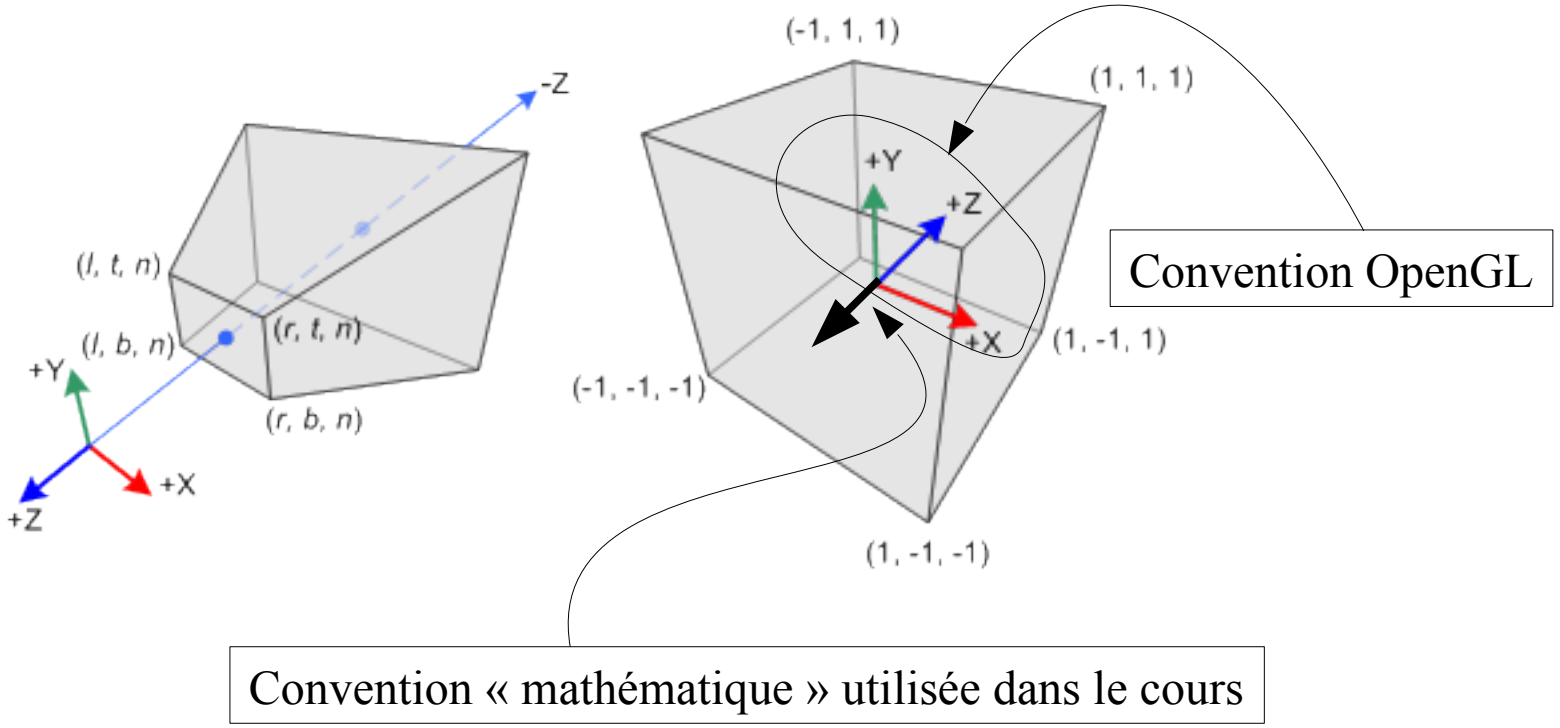
$$z = -|f| \rightarrow z_c = -1$$

Dépend du matériel | Dépend du type de projection | Dépend du point de vue

Infographie

Matrices de transformation

- OpenGL...



Infographie

Matrices de transformation

- En multipliant par -1 et en substituant

$$f<n<0 \quad \text{et} \quad -n=|n| \quad -f=|f| \quad nf=|n||f| \quad n-f=|f|-|n|$$

on obtient :

$$\begin{pmatrix} \frac{2n}{r-l} & 0 & \frac{l+r}{l-r} & 0 \\ 0 & \frac{2n}{t-b} & \frac{b+t}{b-t} & 0 \\ 0 & 0 & \frac{f+n}{n-f} & \frac{2fn}{f-n} \\ 0 & 0 & 1 & 0 \end{pmatrix} \equiv \begin{pmatrix} \frac{-2n}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{-2n}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{-f-n}{n-f} & \frac{2fn}{n-f} \\ 0 & 0 & -1 & 0 \end{pmatrix} \equiv \begin{pmatrix} \frac{2|n|}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2|n|}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{|f|+|n|}{|f|-|n|} & \frac{2|f||n|}{|f|-|n|} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

Matrices de transformation

- Matrice de projection perspective OpenGL conventionnelle

$$\begin{pmatrix} \frac{2|n|}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2|n|}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{|f|+|n|}{|f|-|n|} & \frac{2|f||n|}{|f|-|n|} \\ 0 & 0 & -1 & 0 \end{pmatrix} \rightarrow \underbrace{\begin{pmatrix} \frac{2|n|}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2|n|}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{|f|+|n|}{|n|-|f|} & \frac{2|f||n|}{|n|-|f|} \\ 0 & 0 & -1 & 0 \end{pmatrix}}_{M_{proj\_persp\_OpenGL}}$$

$$z=-|f| \rightarrow z_c=1$$

$$z=-|n| \rightarrow z_c=-1$$

- Idem sans transformation perspective

$$\underbrace{\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & 0 \\ 0 & \frac{2}{t-b} & 0 & 0 \\ 0 & 0 & \frac{2}{n-f} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -\frac{r+l}{2} \\ 0 & 1 & 0 & -\frac{b+t}{2} \\ 0 & 0 & 1 & -\frac{n+f}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_c} = \underbrace{\begin{pmatrix} \frac{2}{r-l} & 0 & 0 & \frac{l+r}{l-r} \\ 0 & \frac{2}{t-b} & 0 & \frac{b+t}{b-t} \\ 0 & 0 & \frac{2}{n-f} & \frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_{proj\_orth}}$$

Infographie

Matrices de transformation

- Projection orthographique OpenGL

$$M_{proj\_orth} = \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & \frac{l+r}{l-r} \\ 0 & \frac{2}{t-b} & 0 & \frac{b+t}{b-t} \\ 0 & 0 & \frac{2}{n-f} & \frac{f+n}{f-n} \\ 0 & 0 & 0 & 1 \end{pmatrix} \rightarrow M_{proj\_orth\_OpenGL} = \begin{pmatrix} \frac{2}{r-l} & 0 & 0 & \frac{l+r}{l-r} \\ 0 & \frac{2}{t-b} & 0 & \frac{b+t}{b-t} \\ 0 & 0 & \frac{-2}{\lvert f\rvert-\lvert n\rvert} & \frac{\lvert f\rvert+\lvert n\rvert}{\lvert f\rvert-\lvert n\rvert} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Dans tous les cas :

$$M = M_s \cdot M_{proj} \cdot M_v$$

Matrices de transformation

- Simplifications habituelles
  - En général, on regarde le centre du volume

$$[l,r]\times[b,t]\times[f,n]$$

$$r=-l=\frac{w}{2} \qquad t=-b=\frac{h}{2}$$

On veut aussi des pixels carrés $\frac{w}{h}=\frac{n_x}{n_y}$

$$M_{proj\_orth\_OpenGL}=\begin{pmatrix} \frac{1}{w} & 0 & 0 & 0 \\ 0 & \frac{n_x}{w\,n_y} & 0 & 0 \\ 0 & 0 & \frac{-2}{|f|-|n|} & \frac{|f|+|n|}{|f|-|n|} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$M_{proj\_persp\_OpenGL}=\begin{pmatrix} \frac{|n|}{w} & 0 & 0 & 0 \\ 0 & \frac{|n|\,n_x}{w\,n_y} & 0 & 0 \\ 0 & 0 & \frac{|f|+|n|}{|n|-|f|} & \frac{2|f||n|}{|n|-|f|} \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

Matrices de transformation

- On peut aussi spécifier le champ de vision (pour les vues en perspective uniquement)

$$\tan\theta = \frac{w}{|n|} \Rightarrow w = |n|\tan\theta$$

$$\underbrace{\begin{pmatrix} \frac{1}{\tan\theta} & 0 & 0 & 0 \\ 0 & \frac{n_x}{n_y \tan\theta} & 0 & 0 \\ 0 & 0 & \frac{|f|+|n|}{|n|-|f|} & \frac{2|f||n|}{|n|-|f|} \\ 0 & 0 & -1 & 0 \end{pmatrix}}_{M_{proj\_persp\_OpenGL}}$$

- Exercice
  - Matrice globale de transformation pour un observateur situé en (x=10,y=10,z=10) qui regarde dans la direction (-1,-1,-1). Un vecteur du plan vertical est le vecteur (0,1,0). L'écran fait 1000 par 1000 pixels.
  - Angle de vue : 45° (tan 45 = 1)
  - Position des plans n et f : 10 et 20

$$w = -\frac{g}{\|g\|} \qquad u = -\frac{t \times w}{\|t \times w\|}$$

$$v = w \times u$$

$$\underbrace{\begin{pmatrix} \frac{n_x}{2} & 0 & 0 & \frac{n_x - 1}{2} \\ 0 & \frac{n_y}{2} & 0 & \frac{n_y - 1}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_s}$$

$$\underbrace{\begin{pmatrix} \frac{2|n|}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2|n|}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & \frac{|f|+|n|}{|n|-|f|} & \frac{2|f||n|}{|n|-|f|} \\ 0 & 0 & -1 & 0 \end{pmatrix}}_{M_{proj\_persp\_OpenGL}}$$

$$\underbrace{\begin{pmatrix} x_u & y_u & z_u & 0 \\ x_v & y_v & z_v & 0 \\ x_w & y_w & z_w & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -x_o \\ 0 & 1 & 0 & -y_o \\ 0 & 0 & 1 & -z_o \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{M_v}$$

Matrices de transformation

- Technique de « sélection »
  - Utilisation de l'inverse des transformations
  - Nécessité de connaître pour chaque pixel, la primitive géométrique dernièrement tracée
    - On verra comment plus tard...

Deux paradigmes pour la génération d'images

Infographie

2 paradigmes...

- Par projection des objets sur le plan de l'écran
  - Aspects purement géométriques
    - Utilisation de matrices de transformation
    - Besoin d'algorithmes d'élimination de faces cachées
    - Techniques de « clipping » (découpage) et « culling » (abattage)
      - Permet de ne tracer que le strict nécessaire en évitant les « effets de bord »
  - Coloriage / ombrage
    - Éclairage
    - Textures
    - Lois de réflexion
  - Possibilité de graphisme en temps réel
  - Implémentation type OpenGL (hardware)

Infographie

2 paradigmes...

- Organigramme
  - Partir des objets et leurs coordonées dans l'espace
  - Déterminer en chaque point (sommet) ou sur chaque facette des caractéristiques d'éclairage, de textures, etc...
  - Projeter dans l'espace des coordonnées de l'écran
    - Matrices de transformation
  - Dessiner l'objet sous forme discrétisée
    - « Rasterisation » (!) pb d'aliasing
    - Tenir compte des faces cachées !
    - La couleur de chaque pixel est déterminée par l'information calculée plus haut

Infographie

2 paradigmes...

- Par lancer de rayon (ray-tracing)
  - Aspects géométriques
    - On part des pixels pour aller à la rencontre des objets
      - Recherche de l'intersection d'un rayon et des objets (habituellement triangulés)
    - Problème des faces cachées : automatiquement réglé !
    - Culling utile (pour éviter les recherches inutiles)
  - Coloriage
    - Lois réalistes de réflexion
    - Lois réalistes d'éclairement
    - Textures complexes
  - Relative lenteur mais haute fidélité
  - Introduction de nouveaux modèles aisée

Infographie

2 paradigmes...

- Organigramme
  - Partir des pixels de l'écran
  - Tracer une ligne passant par l'œil et ce pixel et calculer l'intersection avec le premier objet rencontré
    - Cas particulier si la surface est un diélectrique ou un métal (métal lisse ou verre, liquide etc...) : le rayon est réfléchi, simplement dévié ou se multiplie par réflexion partielle
  - Déterminer la couleur de l'objet en ce point
    - Utilisation de lois physiques ou empiriques
    - Tenir compte de l'éclairement - et des ombres
  - Donner au pixel la couleur déterminée
    - (!) de nouveau, problèmes d'aliasing possibles !

Infographie

2 paradigmes...

- Sujets communs aux deux techniques
  - Lois de réflexion, types de surfaces
  - Textures
  - Modélisation surfacique
  - Maillages
  - Théorie de la couleur

Infographie

2 paradigmes...

- Sujets spécifiques à l'approche par lancer de rayon
  - Equation de radiosité
  - Calcul d'ombres
  - Réfraction, réflexions

Infographie

2 paradigmes...

- Sujets spécifiques à l'approche par projection
  - Problèmes de visibilité (clipping et faces cachées)
  - « Rasterisation »

Infographie

2 paradigmes...

- Il existe des approches hybrides
  - Permet de cumuler les avantages des deux paradigmes
    - Vitesse (approche par projection)
    - Fidélité (approche par lancer de rayon)
  - E.g. Pixar Renderman.

Projet

Infographie

Projet

- Proposition de sujets
  - A) Homemade Ray Tracer (informatique)
    - Faire évoluer un programme de ray-tracing (que je vous fournis)
    - Programmation & implémentation (cf suite)
  - B) Rendu réaliste (architectes)
    - Utiliser Blender afin d'effectuer un rendu *réaliste* d'une scène architecturale (extérieur cette année) – en partant du projet que vous ferez dans le cadre du cours de P. Leclerc
    - Utilisation approfondie du logiciel au complet
  - C) Sujet au choix
    - Si vous voulez travailler sur un sujet qui vous intéresse et qui rentre dans le cadre du cours (je dois le valider *à priori*)

Infographie

Projet

- Déroulement du projet
  - Groupes de deux maximum
  - Délivrables
    - (A) Homemade Ray Tracer
      - a) « cahier des charges » : définition de ce que vous allez faire (+ répartition) , ...
      - b) Rapport final détaillant la philosophie de votre contribution (le pourquoi du comment) + code + résultats et analyse critique + éventuellement détail des contributions personnelles si vous êtes deux
      - c) Petite présentation pour illustrer le fonctionnement de votre code
    - (B) Rendu réaliste
      - a) « cahier des charges » : description du bâtiment, niveau de détail prévu, quel genre de scène, ...
      - b) Rapport final détaillant sous forme de tutoriel comment vous êtes arrivés au résultat + fichiers + résultat et analyse critique
      - c) Présentation (même temps que le cours de MAO)

Infographie

Projet

- (C) Sujet au choix
  - a) Définition du sujet (originalité, intérêt, contexte ...)
  - b) Planification de la réalisation (« cahier des charges »)
    - Si possible, rendre cela en même temps que a)
  - b) rapport final (contenu à définir lors de la validation du sujet) + éventuellement détail des contributions personnelles si vous êtes deux
- Limites physiques des documents
  - Sujet (c) : 1 page A4
  - Cahier des charges : 1-2 pages A4
  - Rapport final : max 15 pages A4

Projet

- Deadlines
  - Choix des sujets / formation des groupes : 7 Mars
    - Dans le cas C) remise du sujet : 7 Mars
  - Remise des cahiers des charges : 21 Mars
  - Remise des rapports : avant le 1er Juin
    - Architectes : lors de la soutenance (ou immédiatement après)
  - Poids relatif de l'exercice : environ 40%

Infographie

Projet

- Homemade Ray-tracer
  - Canevas disponible sur le site du cours

    http://www.cgeo.ulg.ac.be/infographie
  - Développé en C++ sous linux de façon portable
  - Utilisation de Opengl et FLTK
    - Je ne demande pas de développer une interface graphique significative pour ce que vous faites (prend trop de temps) – simplement d'implémenter les modèles de réflexion , ombres, géométrie etc... et de tester directement dans le code (dans la fonction main() par exemple)
    - Le canevas est peu commenté...
    - Vous avez le droit de modifier l'existant...

Infographie

Projet

- 8 sujets
  - 1 – Effets volumiques
    - Fumée, solides translucides, (subsurface scattering?)
  - 2 – Représentation d'objets sans épaisseur
    - Lignes / courbes 1d , points 0d
    - Cheveux, etc..
  - 3 – Modélisation de sources lumineuses complexes
    - Sources lumineuses étendues et visibles dans le champ

Infographie

Projet

4 – Modélisation procédurale

- Calcul de maillages correspondant à des paysages ou des formes vivantes (arbres, etc...)

5 – Modélisation de la profondeur de champ

- Flou artistique lié à l'ouverture de l'objectif

6 – Interfaçage avec un modeleur solide

- Fichiers type CATIA, Solidworks, etc.

7 – Algorithme de Metropolis

- Rendu physique non biaisé

8 – Animations

- Calcul de scènes successives et interpolation du mouvement

9* – Rassembler tout !

- Garder en tête que vous travaillez ensemble...
- Une partie de la note du projet dépend de l'intégration de tous les développements.
- Cas du sujet 7 particulier (partir de zéro?)