



Vision Artificielle

Ecole Nationale Supérieure d’Informatique
pour l’Industrie et l’Entreprise

David Roussel – Mars 2012

Florent Chavand – Mars 2007

Vision artificielle

- Objectif
 - Présenter les différents concepts, outils mis en œuvre en vision artificielle, ainsi que les principales applications
- Partie I
 - Traitement numérique des images
- Partie II
 - Vision 3D

Traitement numérique des images (plan 1)

- Introduction à la vision artificielle
- Acquisition des images
 - vision humaine
 - différents types d'images et capteurs
 - caméras vidéo
 - acquisition, numérisation, caractéristiques
- Traitements sur les images
 - analyses statistiques
 - transformées
 - filtrages
 - restauration
 - opérateurs de morphologie

Traitement numérique des images (plan 2)

- Analyse des images
 - extractions de primitives
 - segmentation: recherche de contours / régions / points d'intérêt
 - analyse du mouvement
 - ...
- Modélisation, reconnaissance de formes, localisation 2D
 - descripteurs de contours, régions, points d'intérêt
 - modélisation paramétrique / structurelle
 - reconnaissance de formes
 - localisation 2D
- Transmission/enregistrement d'images
 - compression
 - codage de source, codage de canal

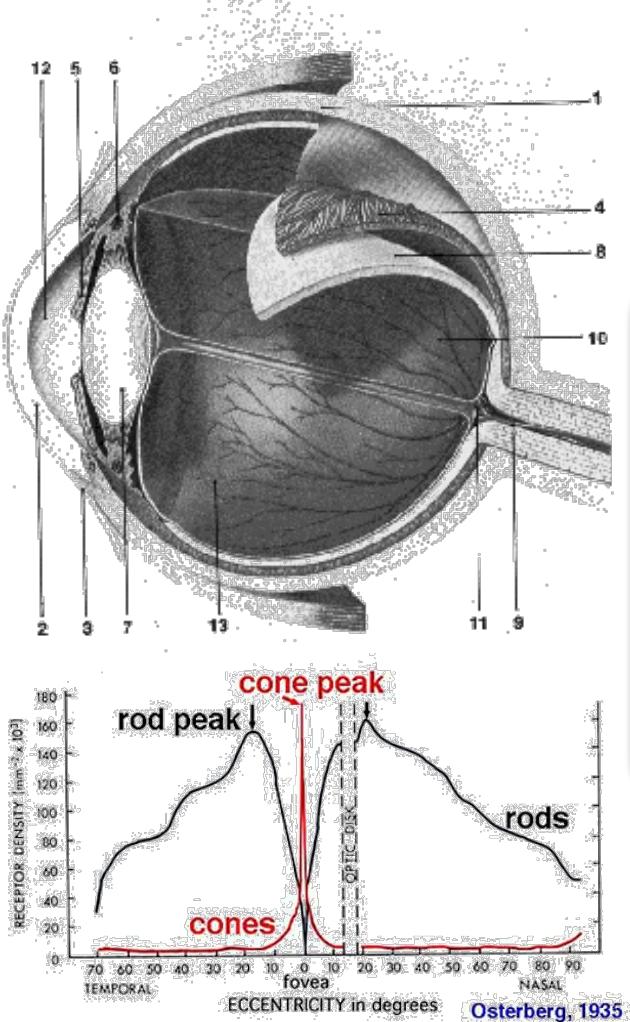
Introduction à la vision artificielle

- Fonctions
 - acquérir, stocker, transmettre des images
 - améliorer
 - extraction d'informations
 - reconnaissance de formes
 - localisation d'objets
 - compression
 - synthèse d'images
- Applications
 - Transmission/enregistrement
 - codage, compression
 - Indexation de documents multimédia
 - reconnaissance de formes, recherches de primitives
 - Donner un sens à une image, à une séquence d'image

Vision artificielle

- Applications (suite)
 - Contrôle d'environnement, d'organes, de systèmes, de process
 - détection, métrologie
 - reconnaissance de formes
 - localisation d'objets, de formes...
 - Suivi de mouvements 2D/3D
 - relevés 3D, réalité virtuelle et augmentée, ...
- Domaines
 - télédétection par satellite
 - médical (échographie, ...)
 - contrôle de process industriels/services
 - télé-surveillance, sécurité
 - jeux, création de films, éducation ('edutainment')
 - télévision, diffusion d'images
 - ...
- Du TDSI au Traitement de l'information

Perception d'une image : l'œil humain



- 1. **Sclérotique** : enveloppe fibreuse avec les muscles moteurs au dessus et en dessous.
- 2. **Cornée** : dans la continuité de la sclérotique
- 3. **Conjonctive** : intérieur de la paupière
- 4. **Choroïde ou Uvée** : zone vasculaire qui irrigue l'extérieur de la rétine
- 5. **Iris**: extension du procès ciliaire
- 6. **Corps ou procès ciliaires** : muscles qui agissent sur le cristallin, émanation de la choroïde, secrète l'humeur aqueuse située entre le cristallin et la cornée.
- 7. **Cristallin** : change de forme grâce aux muscles du procès ciliaire
- 8. **Rétine** : neuro-sensorielle dans les 4/5 postérieurs (cônes + bâtonnets) ailleurs il n'y a pas d'axones
 - Cônes : cellules sensibles à la couleur (3 types) : ~ 6,400,000
 - Bâtonnets : Cellules sensibles à la luminosité : de 110,000,000 à 125,000,000
- 9. **Nerf optique + artère centrale de la rétine**
- 10. **Fovéa** : zone de concentration des cônes
- 11. **Point aveugle** : zone de convergence des axones qui constituent le nerf optique, il n'y a plus de cellules sensorielles à cet endroit
- 12. **Humeur Aqueuse** : sécrétée par les procès ciliaires
- 13. **Corps vitré** : masse gélatineuse non renouvelée.

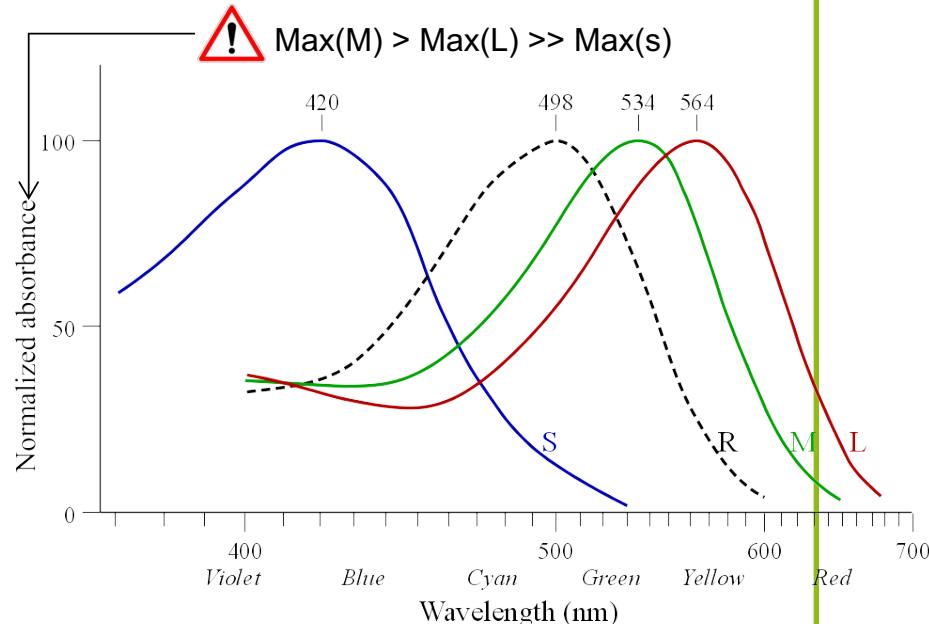
Le système visuel humain: la rétine

- les photorécepteurs: cônes et bâtonnets

- $\sim 6,5 \times 10^6$ cônes
 - vision centrale (fovea), acuité visuelle (120 cônes/ $^\circ$ dans la fovea)
 - temps de réponse = 1/4 temps de réponse des bâtonnets
 - peu sensibles (vision diurne), couleur (3 types: L(rouge-orange), M(vert jaune), S(bleu))
- $\sim 130 \times 10^6$ bâtonnets
 - vision périphérique
 - très sensibles (facteur 100/cônes, vision nocturne), pas sensibles à la couleur

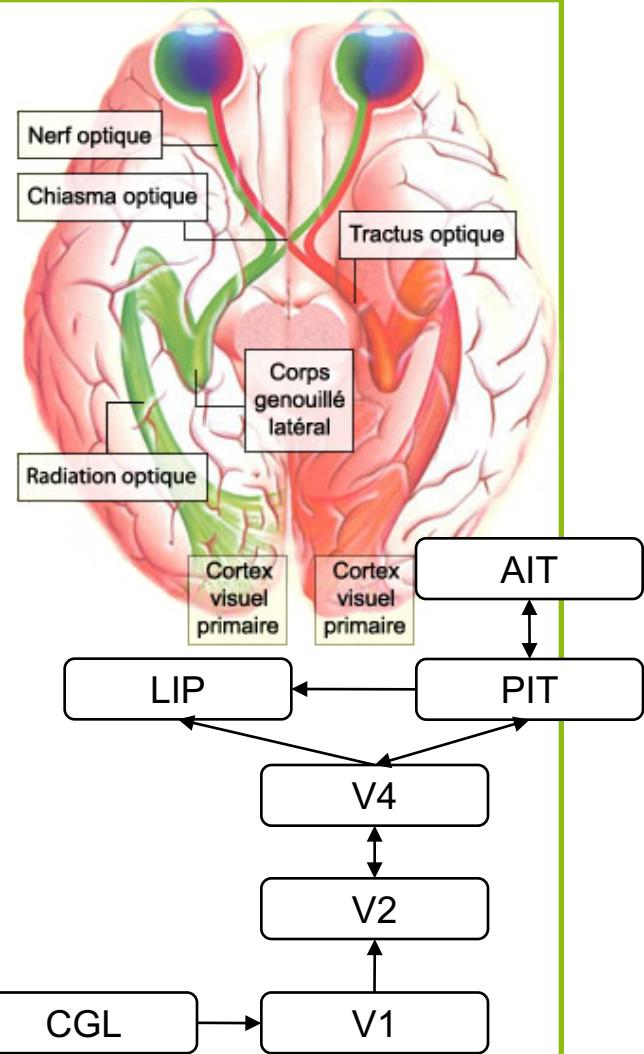
- cellules de pré-traitement des signaux (!!complexe!!)

- horizontales réseau fonctionnel de type spatio-temporel passe bas
- bipolaires: filtrage passe haut de l'image rétinienne
- ...
- ganglionnaires (10^6) axones --> nerf optique



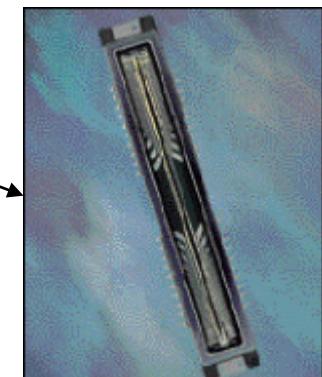
Le système visuel humain (suite)

- corps genouillé du thalamus (10^6 neurones)
- cortex visuel (10^{10} neurones en couches)
 - voie dorsale: analyse du mouvement et de l'espace --> contrôle du geste
 - voie ventrale: reconnaissance des objets (5 niveaux d'analyse)
 - V1 ($500 \cdot 10^6$ neurones), V2: orientations, fréquences spatiales, disparité stéréoscopique, direction du mouvement, couleur.
 - V4: couleurs et formes
 - cortex inférotemporal (PIT, AIT): reconnaissance de formes complexes, de visages,.....; invariance à la position de la forme, à l'éclairage, ..
- couches de neurones hiérarchiquement organisées
 - plus on progresse dans la hiérarchie des aires visuelles, plus la taille des champs récepteurs augmente (AIT ≈ tout le champ visuel), le temps de latence augmente ainsi que l'insensibilité à des stimuli simples.
- temps de latence
 - 150 ms pour analyser une image
- référence
 - Jean-Michel Jolian - *Les systèmes de vision*. Traité des Sciences Pour l'Ingénieur. Ed. Hermès science publications, 2000, ISBN : 2746201852 (chap 2 et 3)



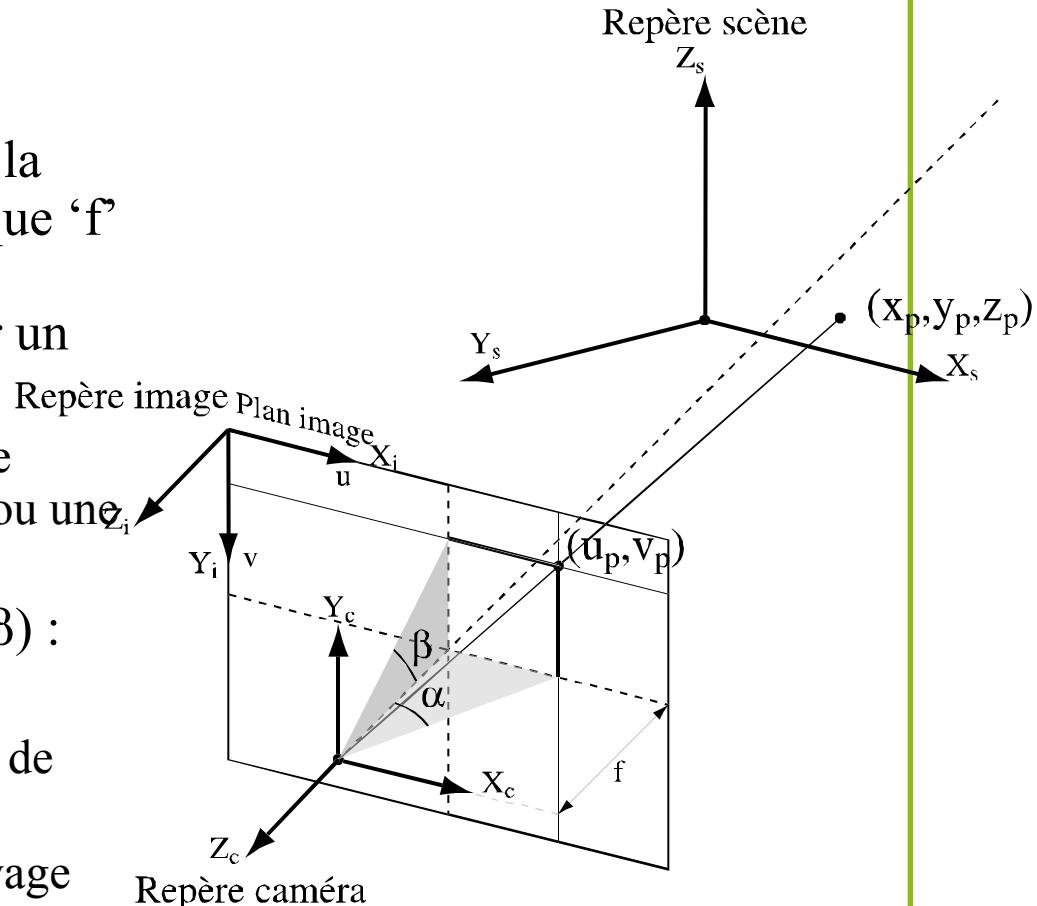
Images et capteurs d'images

- Images
 - de températures, de pression, de luminance
 - de profondeurs, de distances
 - échographiques
 - RMN
- Capteurs
 - Caméras
 - Matrice : 1/3 pouce, 1/2 pouce
 - de 1 Mpixels à 16 Mpixels (4000×4000)
 - Vecteur : caméra linéaire
 - ~10 Kpixels
 - exemple : scanners
 - Technologie
 - intégrateurs de lumière
 - Matrices de photodiodes
 - » CCD : Charge Coupled Device
 - » CMOS : Complementary Metal Oxide Semiconductor
 - Couleur \Rightarrow multiplication des capteurs (R,V,B)
 - Rétines artificielles
 - l'optique est directement intégrée à la matrice

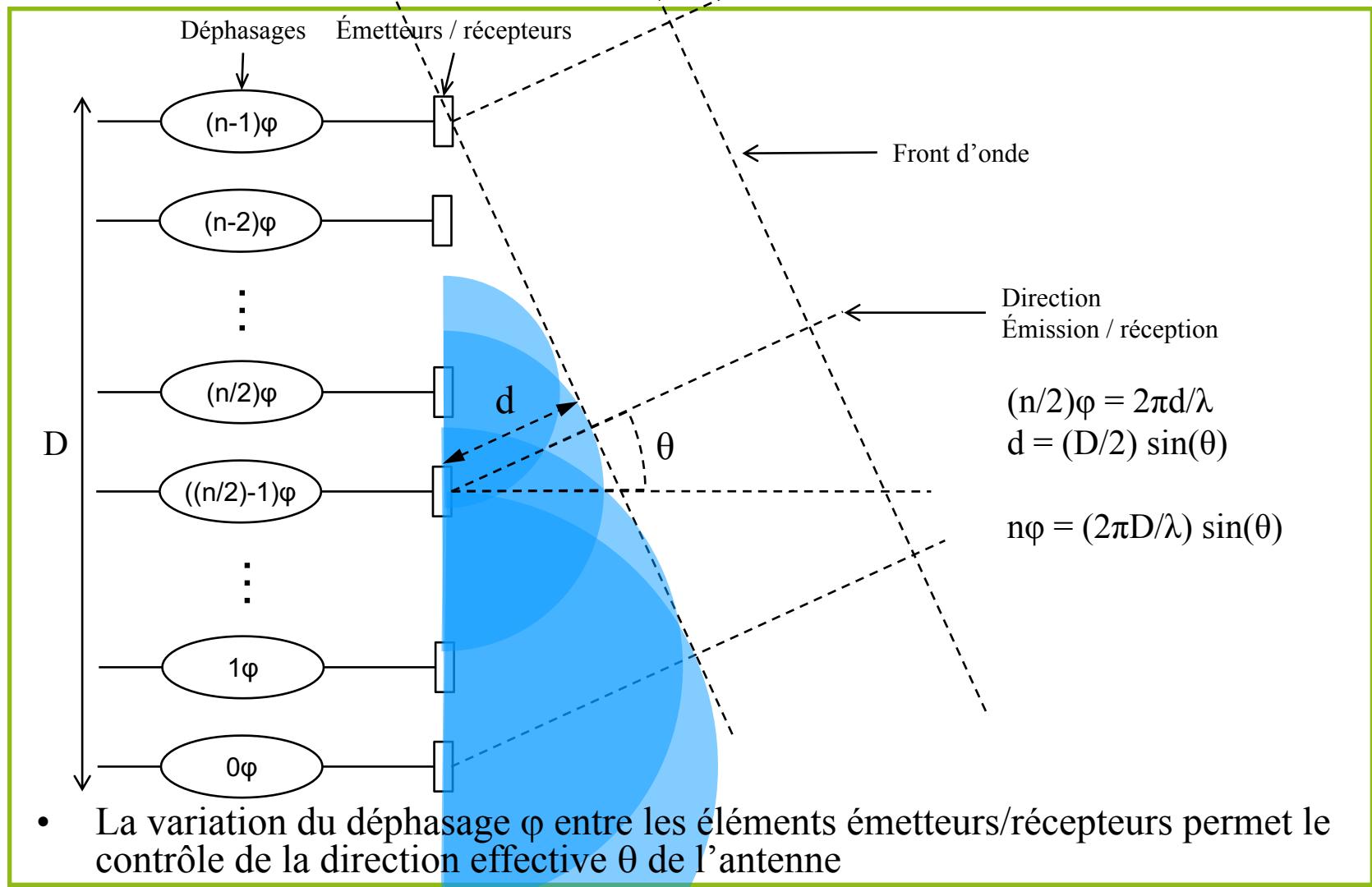


Qu'est ce qu'une image ?

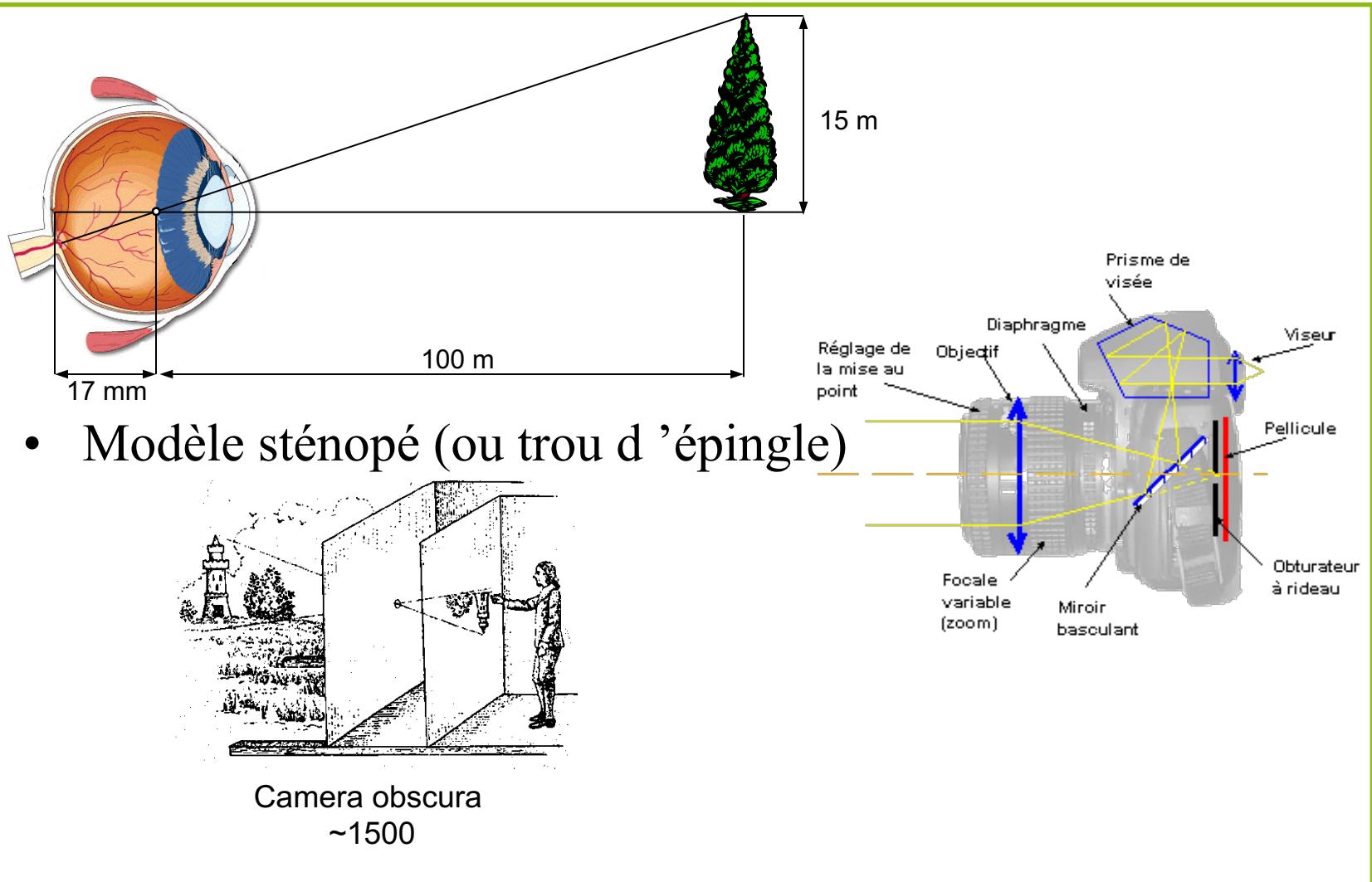
- Imagerie 1D = cas particulier imagerie 2D.
- Image 2D = représentation de la mesure d'une grandeur physique 'f' dans la direction (α, β) avec, éventuellement, projection sur un plan image $f(u, v)$.
 - $f(\alpha, \beta)$ est un éclairement, une température, une absorption ou une distance.
- Pointage dans la direction (α, β) :
 - optique (lentille),
 - mécanique (tourelle 2 degrés de liberté),
 - électronique (antenne à balayage électronique par gestion des déphasages des éléments actifs).



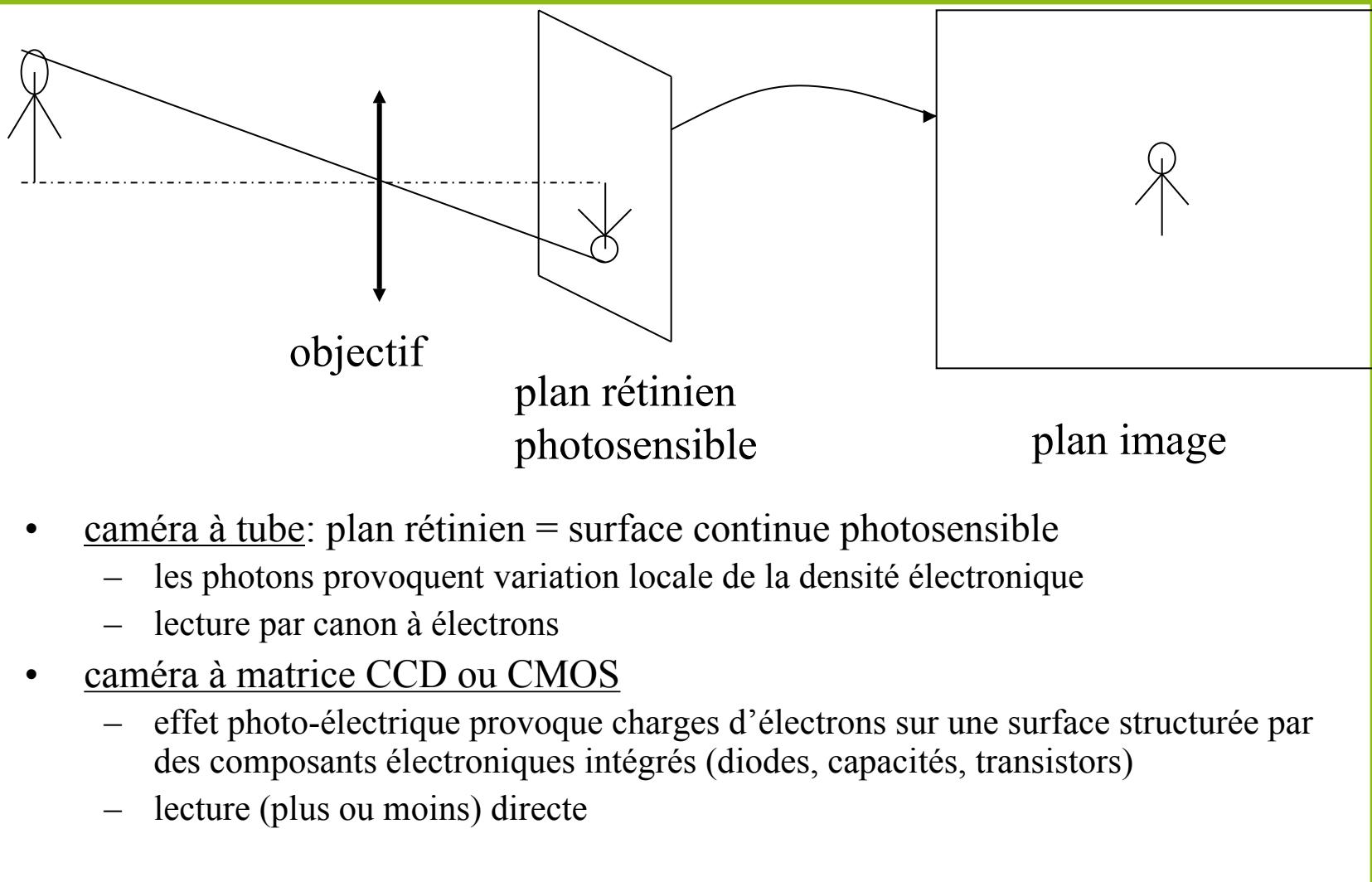
Antenne à balayage électronique



Principes de formation des images

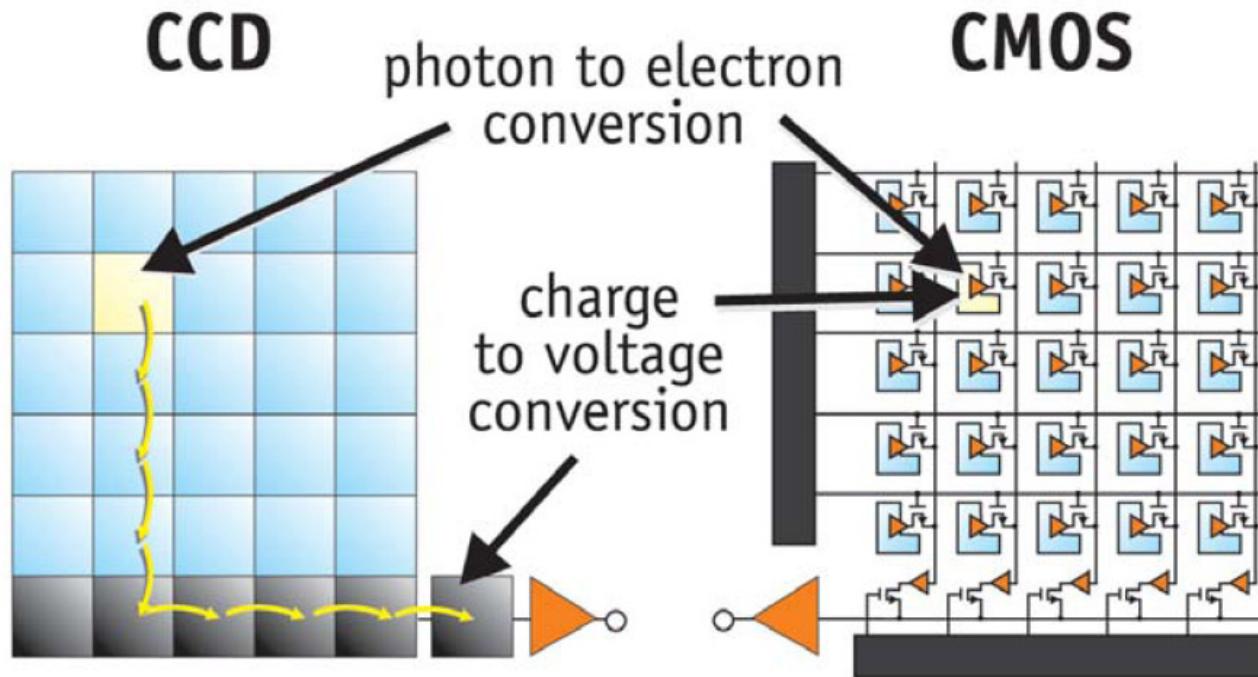


Appareil photo, Caméra vidéo



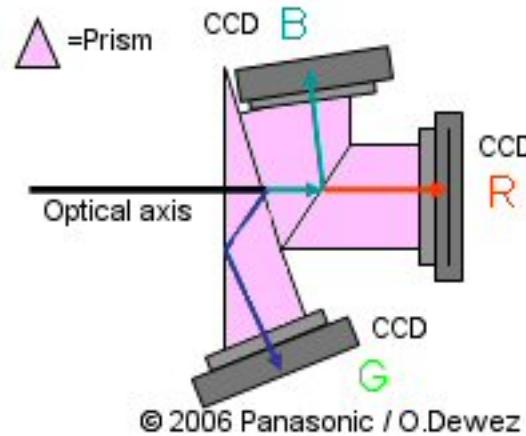
Matrices photosensibles

- CCD : Charge-Coupled Device
 - + Sensibilité plus grande
 - Nécessité d'unités de traitements
- CMOS : Complementary Metal–Oxide–Semiconductor
 - + Plus grande intégration
 - Zone non photosensible des pixels (fill factor)

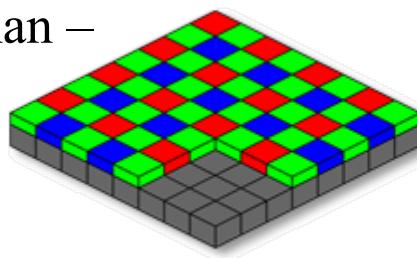
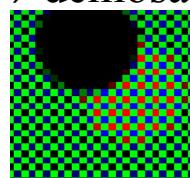


Matrices couleur

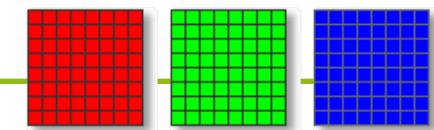
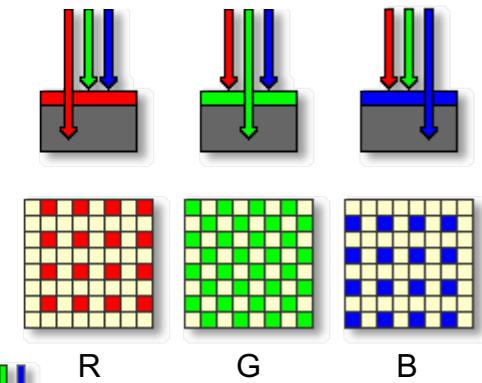
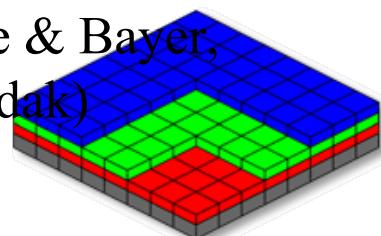
- Multiplier les capteurs :
 - p. ex. Caméras TRI-CCD



- Utiliser un arrangement de filtres de couleurs
 - Filtre de Bayer (Eastman – Kodak)
 - \Rightarrow demosaicing

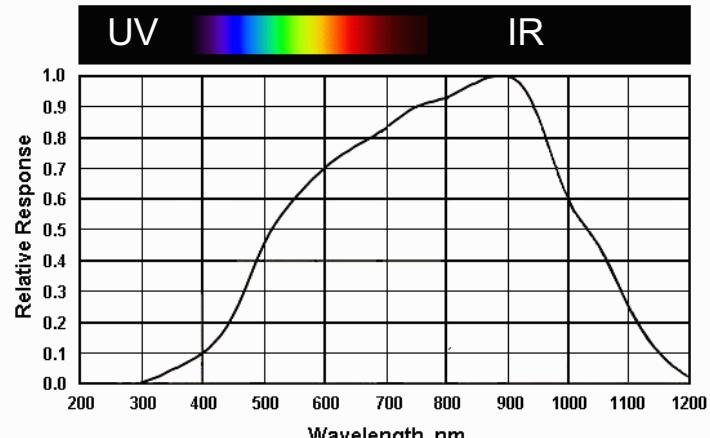
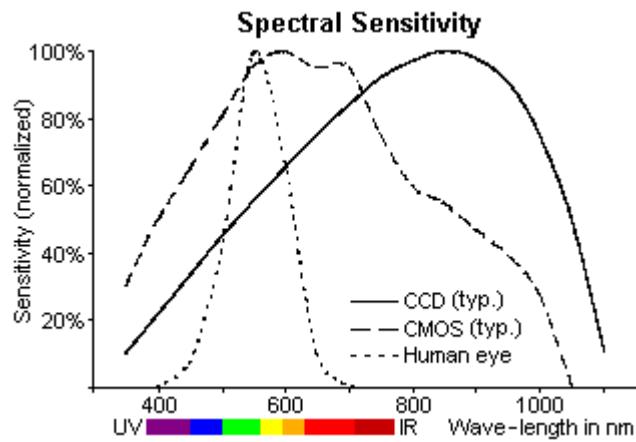


- Foveon (Bryce & Bayer, Eastman – Kodak)

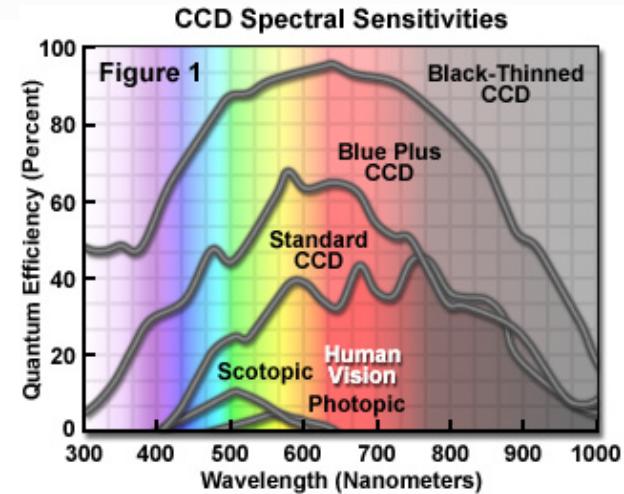


Sensibilité spectrale des capteurs

- La sensibilité du silicium s'étende sur une bande passante plus grande que la sensibilité de l'œil humain
 - Nécessité d'un filtre infrarouge

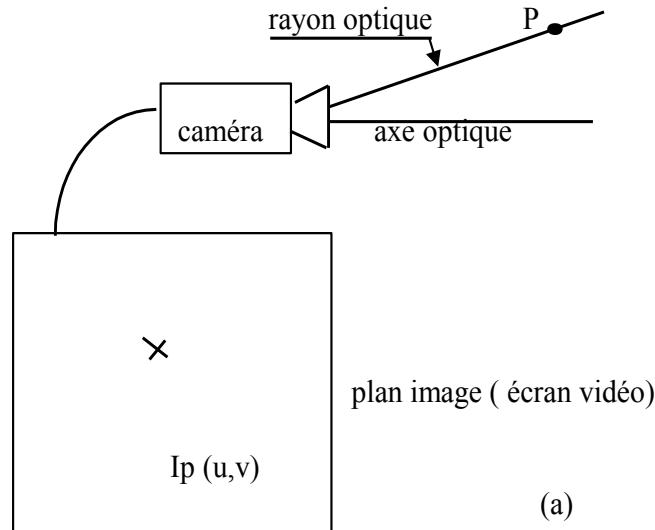


Typical Silicon Photodiode Spectral Response

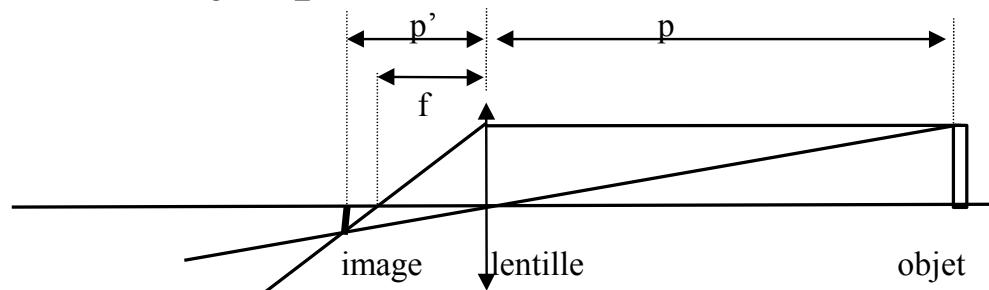


Rayon optique \Rightarrow Pixel dans une caméra

- Rayon optique
 \Rightarrow Pixel dans une caméra

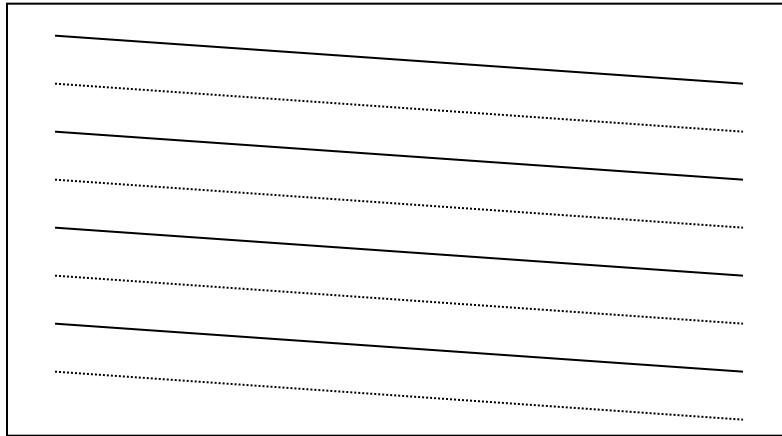


- Image d'un objet par une lentille

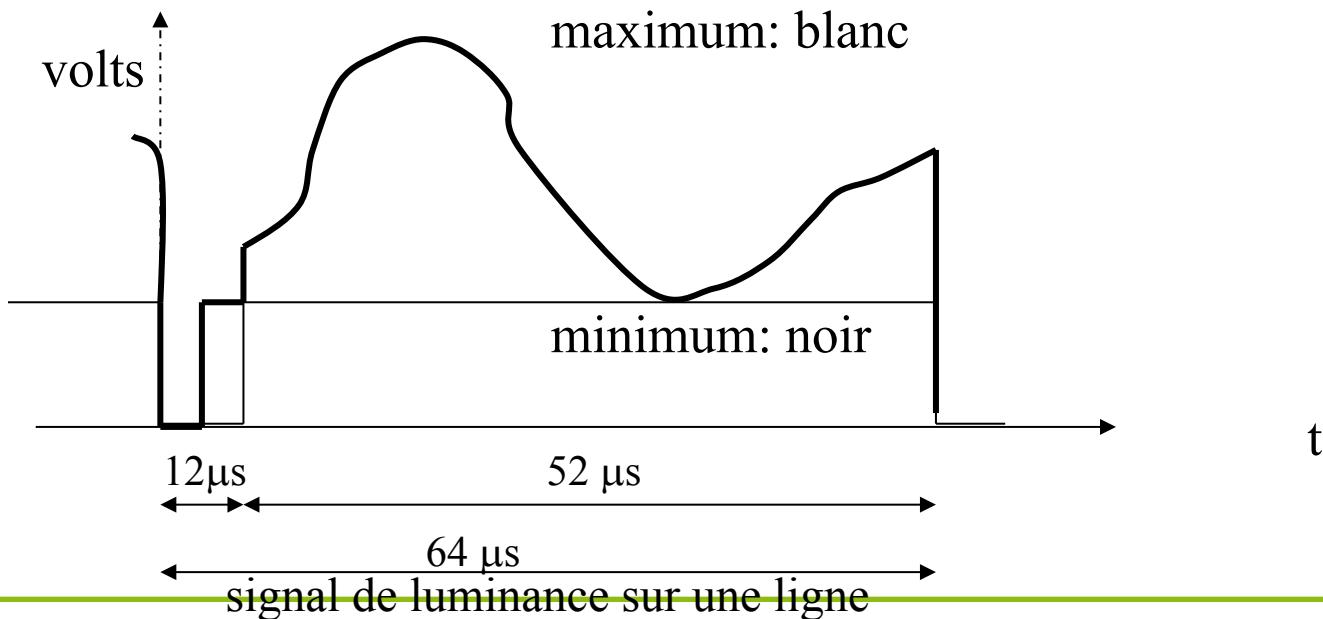


$$\frac{1}{p} + \frac{1}{p'} = \frac{1}{f}$$

Forme du signal vidéo analogique



une image = 2 trames entrelacées
50 trames, 25 images/seconde
625 lignes



Signaux vidéo analogiques

Propriété \ Standard	NTSC	PAL (SD)
images/seconde	29.97	25
ms / image	33.37	40
lignes/image = résolution	525	625
l/h = aspect ratio	4:3	4:3
entrelacement	2:1	2:1
$\mu\text{s} / \text{ligne}$	63.56	64

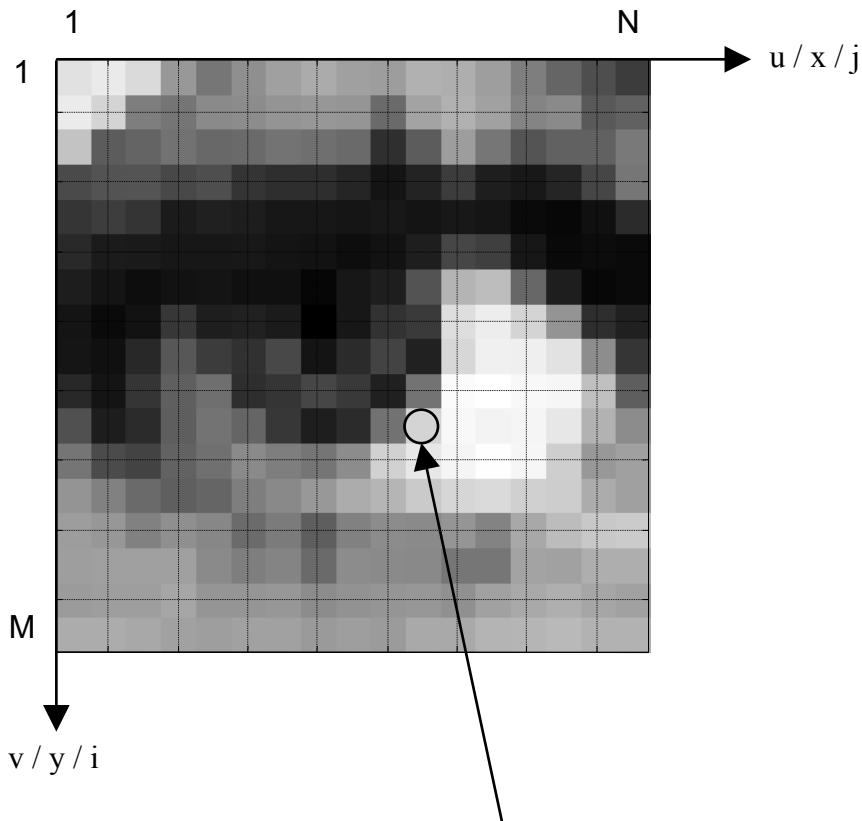
Signaux vidéos numériques

- SD : Standard Definition (interlaced)
 - 4:3
 - NTSC
 - 640×480
 - PAL
 - 720×576 (5:4) rescaled to 768×576 (4:3)
 - 16:9
 - NTSC
 - 704×480 (22:15) rescaled to 848×480 (53:30 \approx 16:9)
 - PAL
 - 720×576 (5:4) rescaled to 1024×576 (16:9)
 - HD : High Definition
 - HD Ready : 1280×720 (16:9)
 - 720p : progressive scan
 - Full HD : 1920×1080 (16:9)
 - 1080i : interlaced
 - 1080p : progressive scan → Full HD
 - 4K (UHD) : initialement 4096 , puis 3840×2160 (16:9) = (1980×2 , 1080×2)
 - 2160p : progressive scan
- Normes de compression
 - MPEG 2
 - Débit SD (720×576) : de 2 à 6 Mb/s
 - Débit HD (1920×1080) : ~18 Mb/s
 - MPEG 4
 - Débit SD (720×576) : de 2 à 3 Mb/s
 - Débit HD (1920×1080) : de 13 à 18 Mb/s

Numérisation des images

- pixels: picture elements
- échantillonnage « naturel »: pixels de la matrice CCD ou CMOS
 - $14,6 \times 10^6$ pixels en 2012 : aspect : $24 \times 36 \Rightarrow$ image : 4672×3120
- échantillonnage norme ‘ télévision numérique ’
 - cf. planche transmission/enregistrement
- échantillonnage pour le traitement des images:
 - 512×512 pixels ou 1024×1024 pixels
- quantification sur 8 bits: noir(0), blanc (255)
- images NB:
 - $512 \times 512 \times 8 = 2 \text{ Mbits}/\text{image} \times 25 \text{ images}/\text{s} \Rightarrow 50 \text{ Mb s}^{-1}$
- images couleurs:
 - 3 capteurs $\Rightarrow 6 \text{ Mb}/\text{image} \times 25 \text{ images}/\text{s} \Rightarrow 150 \text{ Mb}/\text{s}$

Image numérisée: notations



- M lignes, N colonnes
 - $P=M \times N$ pixels / image
 - k nombre de bits de quantification
 - $K=2^k$ niveaux de gris
- exemple:
 - $M=N=512$ pixels
 - $k=8 \Leftrightarrow K=256$
- Image NB:
 - 262 144 pixels/image
 - 2 Mb/image
 - 50 Mb/s à 25 im/s
- Image couleur (RVB)
 - 6 Mb/image
 - 150 Mb/s à 25 im/s

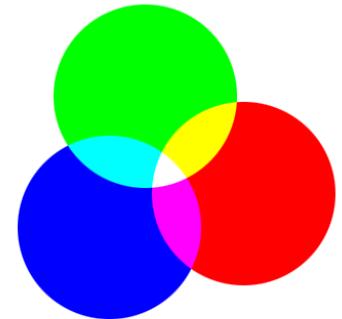
Image numérisée

- Matrice de pixels ($M \times N$)
- Pixel : **Picture Element**
- Types de pixels
 - Noir & Blanc
 - Niveau de gris
 - Couleur
- Etendue des pixels
 - Noir et Blanc : 2 valeurs (2^1)
 - Niveau de gris : 256 valeurs (2^8)
 - Couleur : 256^3 valeurs ($2^{8 \times 3}$)



Codage des pixels (1/2)

- Noir et Blanc : 2 valeurs (2^1)
 - codage : sur 1 bit
 - caractéristique : luminance
- Niveau de gris : 256 valeurs (2^8)
 - codage sur 8 bits \Leftrightarrow 1 octet
 - caractéristique : luminance
- Couleur : $\sim 16M$ valeurs ($2^{8 \times 3}$)
 - codage sur $3 \times$ octets (rouge, vert, bleu)
 - caractéristiques : luminance + chrominance



Codage des pixels (2/2)

- RGB : Red, Green, Blue



Red



Green



Blue

- CMYK : Cyan, Magenta, Yellow, black



Cyan



Magenta



Yellow



Black

- HLS : Hue, Lightness, Saturation



Hue

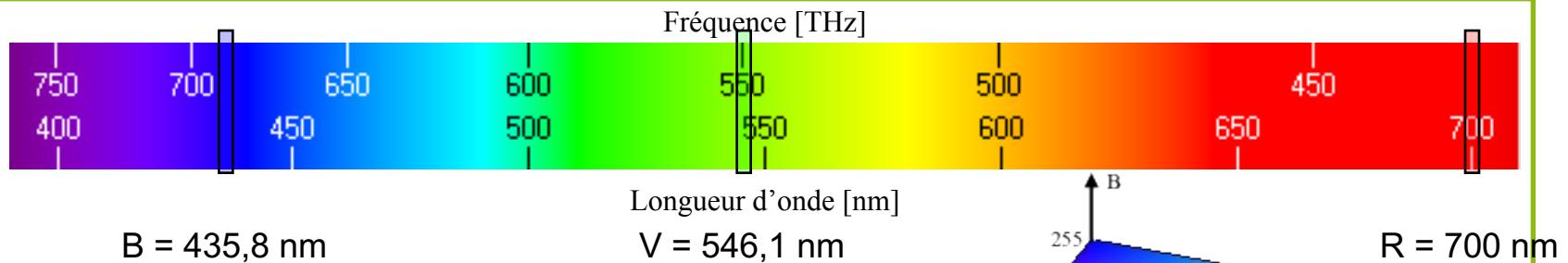


Lightness

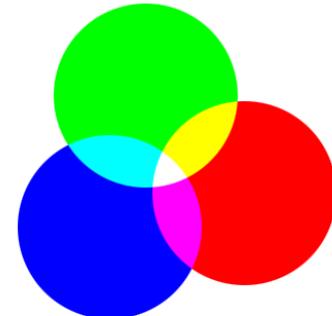
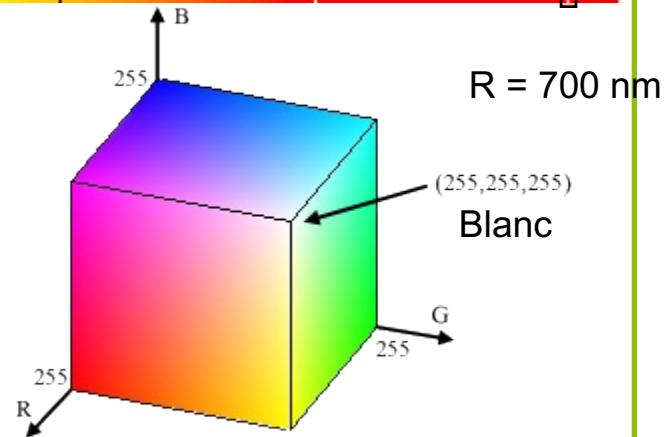


Saturation

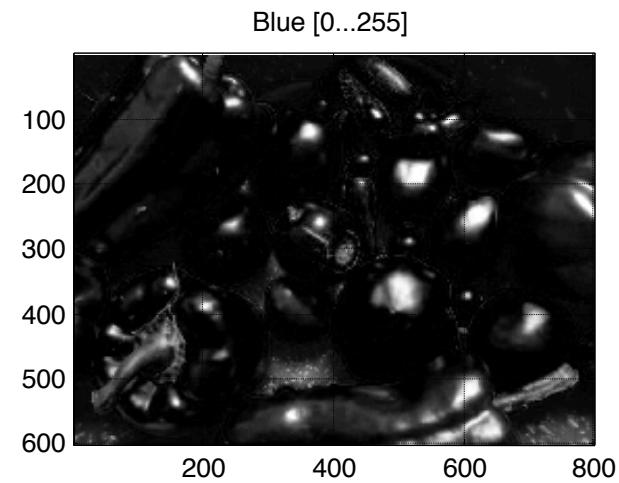
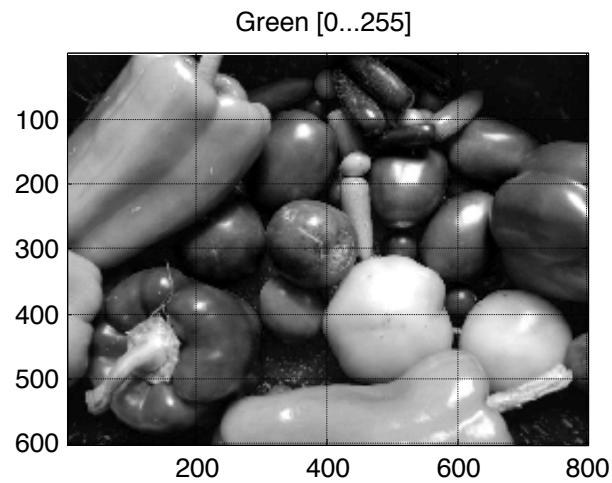
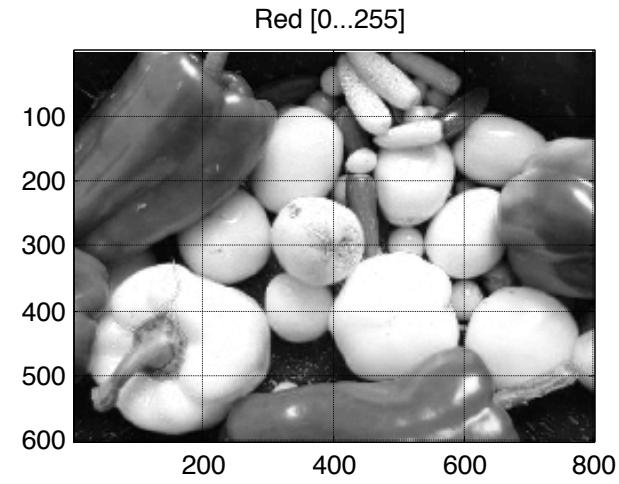
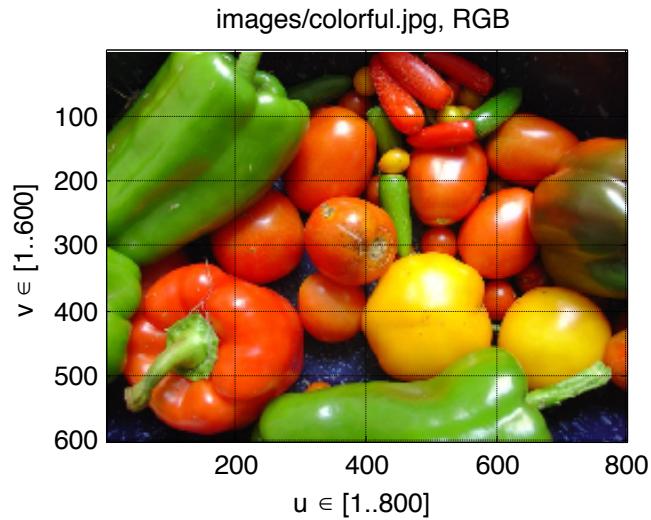
Couleur



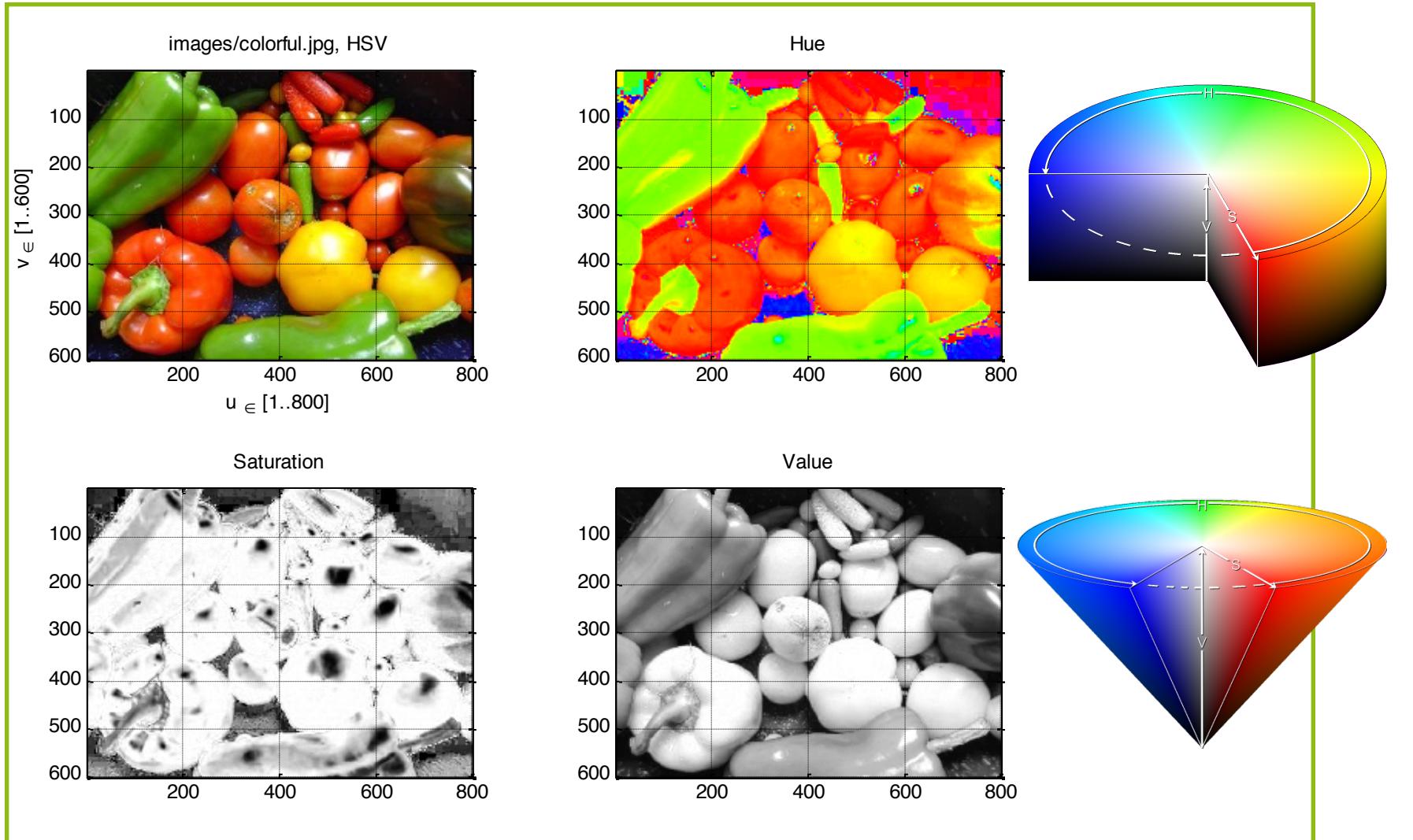
- Synthèse additive de 3 couleurs primaires (RVB)
 - Codage sur $8\text{bits} \times 3 = 24 \text{ bits} = 16 \text{ M couleurs}$
 - Palette (256 couleurs parmi 16M) obsolète.
 - La synthèse additive RVB ne représente qu'une partie des couleurs possibles.



Espace de couleurs RGB (exemple)



Espace de couleurs HSV (exemple)



Espace de couleurs CIE 1931

- Valeurs de Tristimulus
 - L'œil humain possède 3 types de récepteurs (L, M & S) \Rightarrow la sensation de couleur est décrite par 3 paramètres.
 - Dans l'espace CIE 1931 ces paramètres sont X, Y & Z.
 - CIE XYZ est basé sur des mesures de perception visuelle humaine
- Observateur standard
 - Métamérisme : La superposition de deux sources lumineuses de longueurs d'ondes différentes peut apparaître d'une autre couleur.
 - Observateur standard : champ de 2° autour de la fovéa (CIE 1931 2° Standard Observer)
 - L'observateur standard est caractérisé par 3 « color-matching functions ».

W. D. Wright - *A re-determination of the trichromatic coefficients of the spectral colours*. Transactions of the Optical Society, Vol. 30, pages 141-164, 1928

J. Guild - *The colorimetric properties of the spectrum*. Philosophical Transactions of the Royal Society of London, Vol. A230, pages 149-187, 1931

CIE 1931 Color matching functions

- Description numérique de la réponse chromatique de l'observateur
- $\bar{x}(\lambda), \bar{y}(\lambda)$ & $\bar{z}(\lambda)$ représentent les sensibilités spectrales des récepteurs (L, M ou S) qui mènent aux valeurs de tristimulus X, Y & Z.



Modification

$$X = \int_0^{\infty} I(\lambda) \bar{x}(\lambda) d\lambda$$

$$Y = \int_0^{\infty} I(\lambda) \bar{y}(\lambda) d\lambda$$

$$Z = \int_0^{\infty} I(\lambda) \bar{z}(\lambda) d\lambda$$

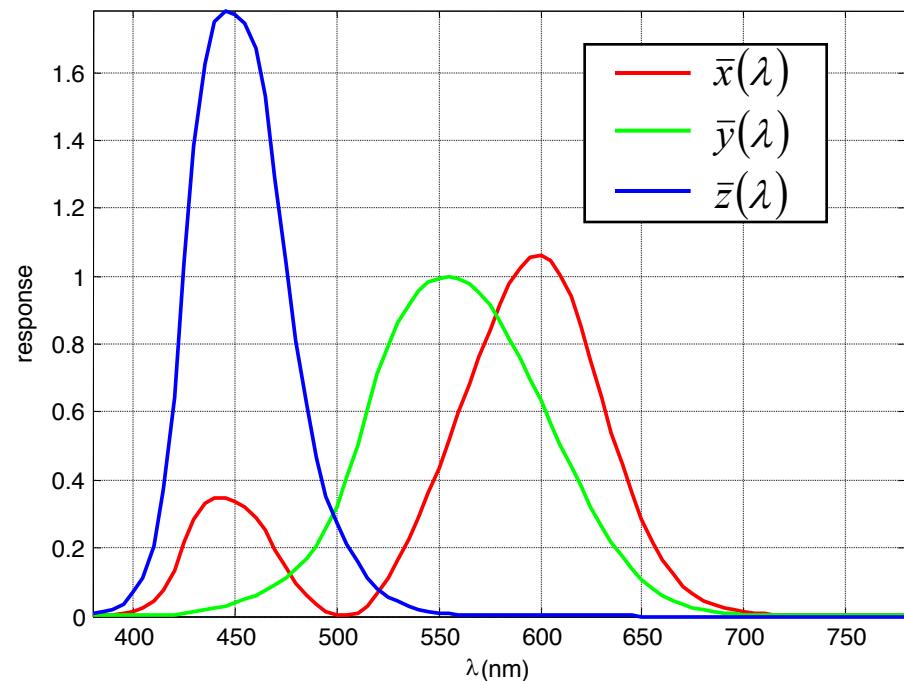
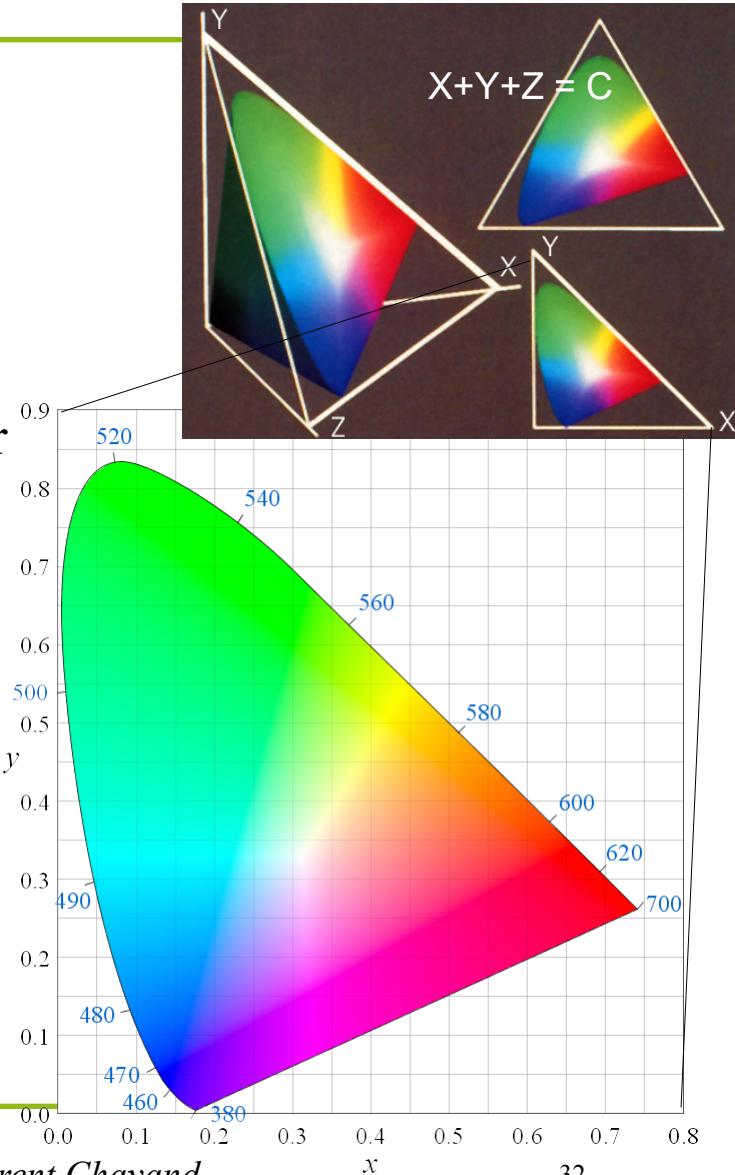


Diagramme de chromaticité

- Couleurs possibles dans XYZ
 - λ dans XYZ
- Couleur = luminance + chrominance ou chromaticité
- Le Y du CIE XYZ peut être considéré comme une mesure de luminance
- La chrominance peut alors être spécifiée par les deux paramètres dérivés x et y
 - Espace de couleur CIE xyY
 - Projection des couleurs possibles sur le plan XY
 - gamut de la vision humaine

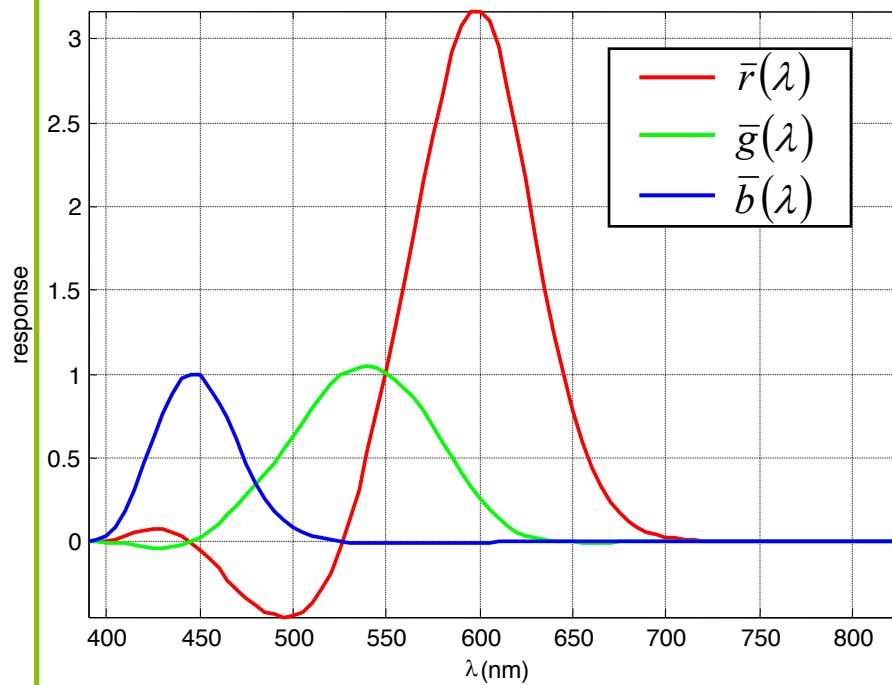
$$x+y+z=1 \quad \left\{ \begin{array}{l} x = \frac{X}{X+Y+Z} \\ y = \frac{Y}{X+Y+Z} \\ z = \frac{Z}{X+Y+Z} = 1-x-y \end{array} \right. \quad \Leftrightarrow \quad \begin{aligned} X &= \frac{y}{y} x \\ Z &= \frac{y}{y} (1-x-y) \end{aligned}$$



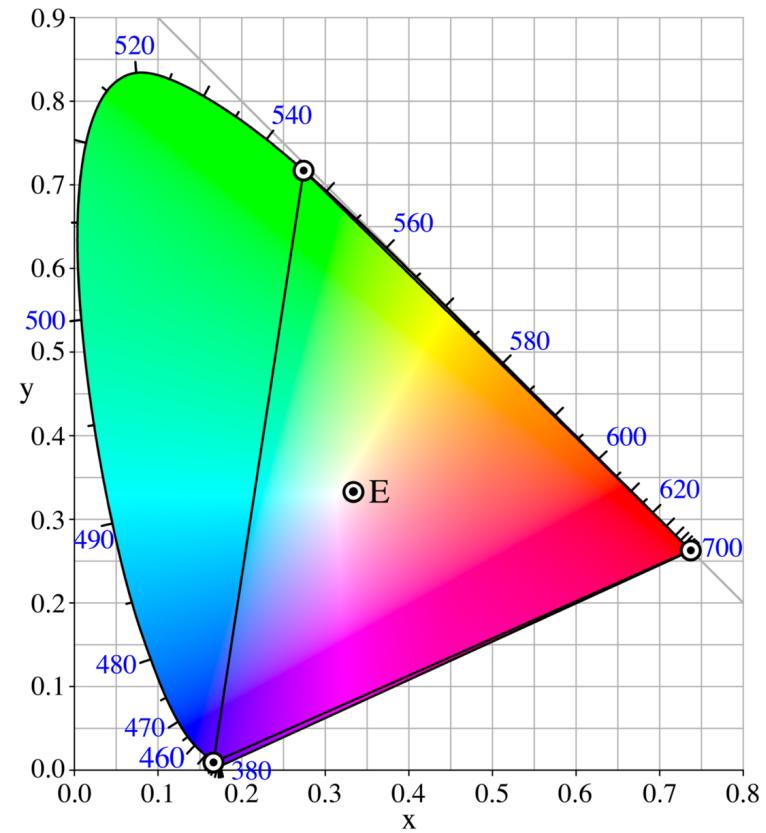
Espace de couleurs CIE RGB

- Dérivé du CIE XYZ

– R = 700 nm
– G = 546.1 nm
– B = 435.8 nm

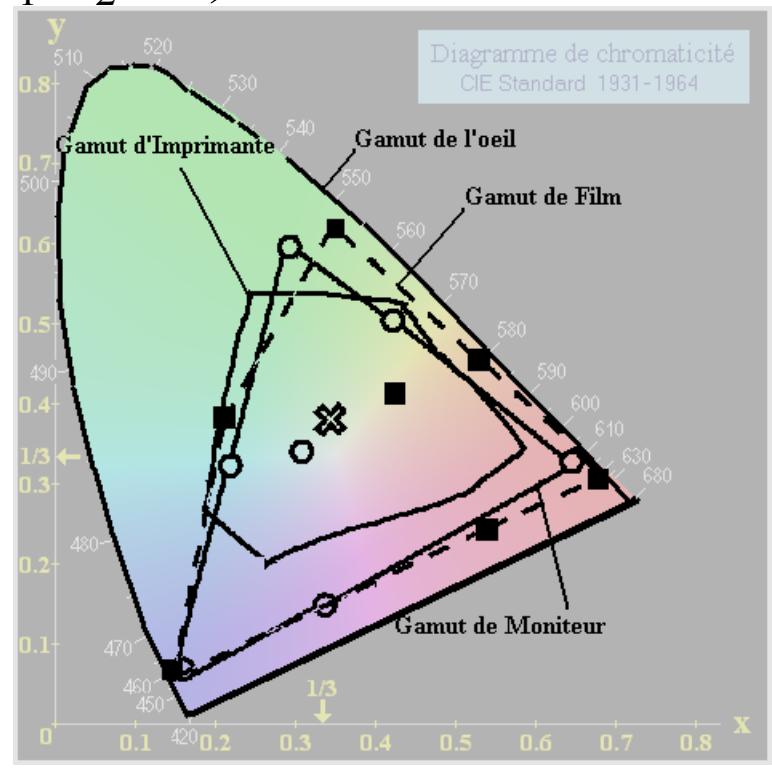
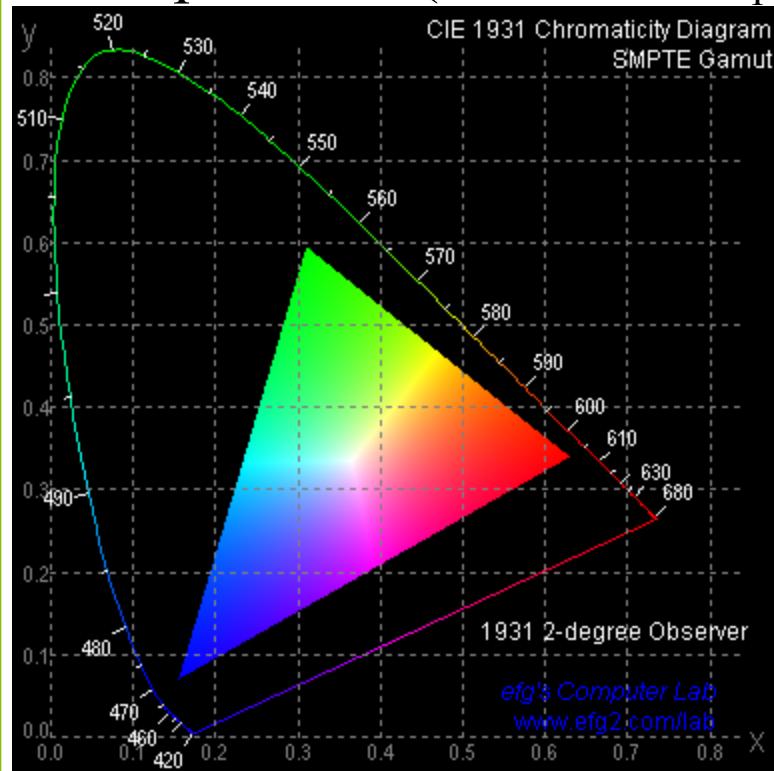


- Gamut de l'espace RGB

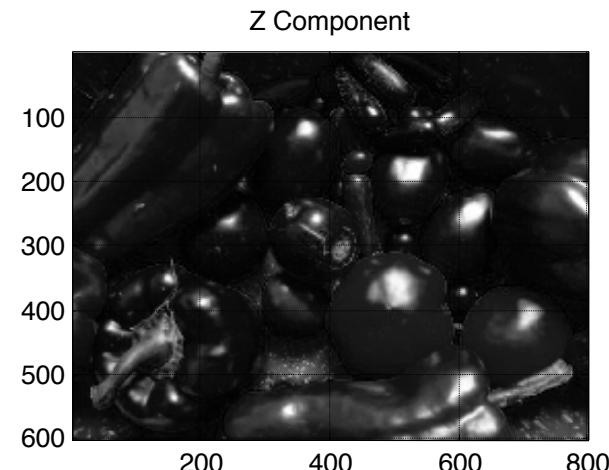
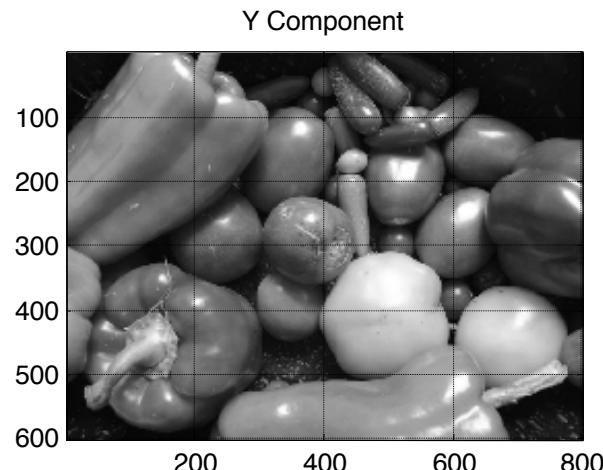
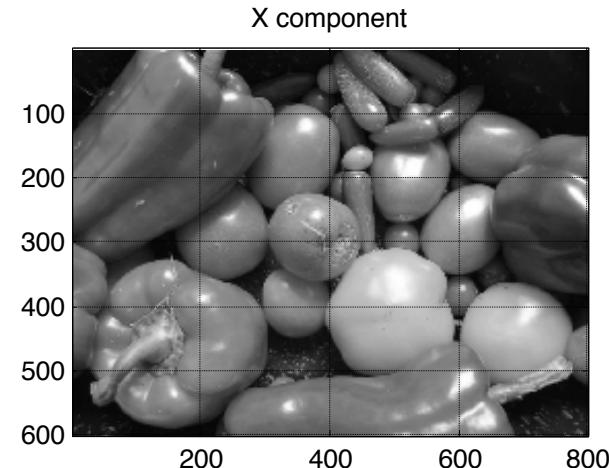
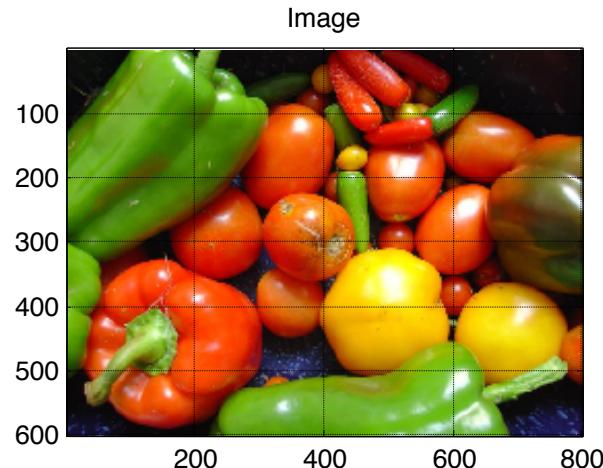


Gamuts

- Film photo (RGB)
- Moniteur (RGB)
- Imprimante ($\text{CMYK} \Leftrightarrow \text{C}_1\text{C}_2\text{M}_1\text{M}_2\text{YK}$)



Espace de couleurs XYZ (exemple)



Numérisation des images fréquences spatiales

- fréquence spatiale maximale présente dans image numérisée:
 - passage du noir au blanc dans deux pixels voisins: $\lambda_s=2$ pixels, $v_s=0.5$ cycle/pixel
- fréquence spatiale minimale:
 - $v_s=0$
 - $\lambda_s=N$ pixels, $v_s=1/N$ cycle/pixel
- Exemple :
 - Image 128×128
 - Motif sinusoïdal 2D:
 - Horizontal : 8 cycles/im $\Leftrightarrow 0.0625$ cycles/pixel
 - Vertical : 4 cycles/im $\Leftrightarrow 0.0313$ cycles/pixel





Analyse statistiques des images et transformations sur les niveaux de gris

Analyse statistique des images

- Grandeurs statistiques globales
 - Image Luminance M lignes \times N colonnes : $P = M \times N$

- Moyenne des niveaux de gris :
$$\bar{m} = \frac{1}{P} \sum_{i=1}^M \sum_{j=1}^N L(i, j)$$

- Variance des niveaux de gris :
$$\sigma^2 = \frac{1}{P} \sum_{i=1}^M \sum_{j=1}^N (L(i, j) - \bar{m})^2$$

- Dispersion des niveaux dans l'image :
$$\frac{\bar{m}}{\sigma}$$

Analyse statistique des images

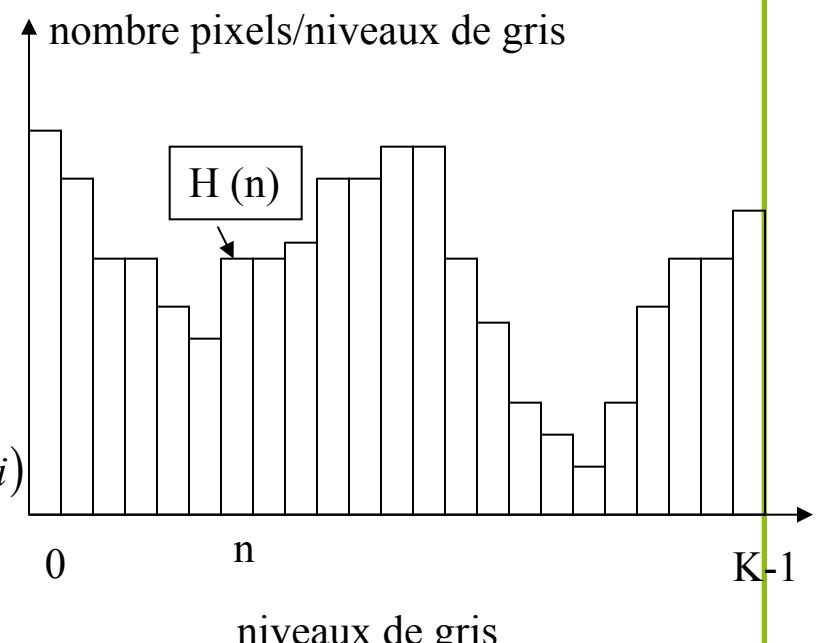
- Histogrammes d'une image de P pixels à K niveaux de gris

$$\sum_{n=0}^{K-1} H(n) = P$$

Fréquence d'apparition :
$$\begin{cases} p(n) = \frac{H(n)}{P} \\ \sum_{n=0}^{K-1} p(n) = 1 \end{cases}$$

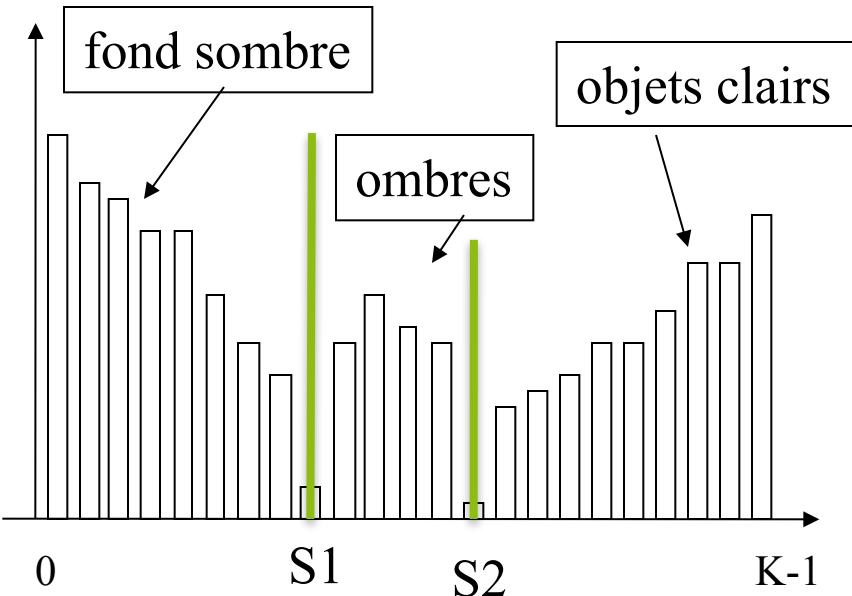
Histogramme cumulé :
$$\begin{cases} H_c(n) = \sum_{q=0}^n H(q) \\ H_c(0) = H(0) \\ H_c(K-1) = P \\ H_c(i) = H_c(i-1) + H(i) \\ \text{avec } 1 < i < K-1 \end{cases}$$

"probabilité" cumulée :
$$\begin{cases} p_c(n) = \sum_{q=0}^n p(q) \\ p_c(K-1) = 1 \end{cases}$$



Transformation et amélioration de l'image

- Binarisation de l'image \Rightarrow Seuillage

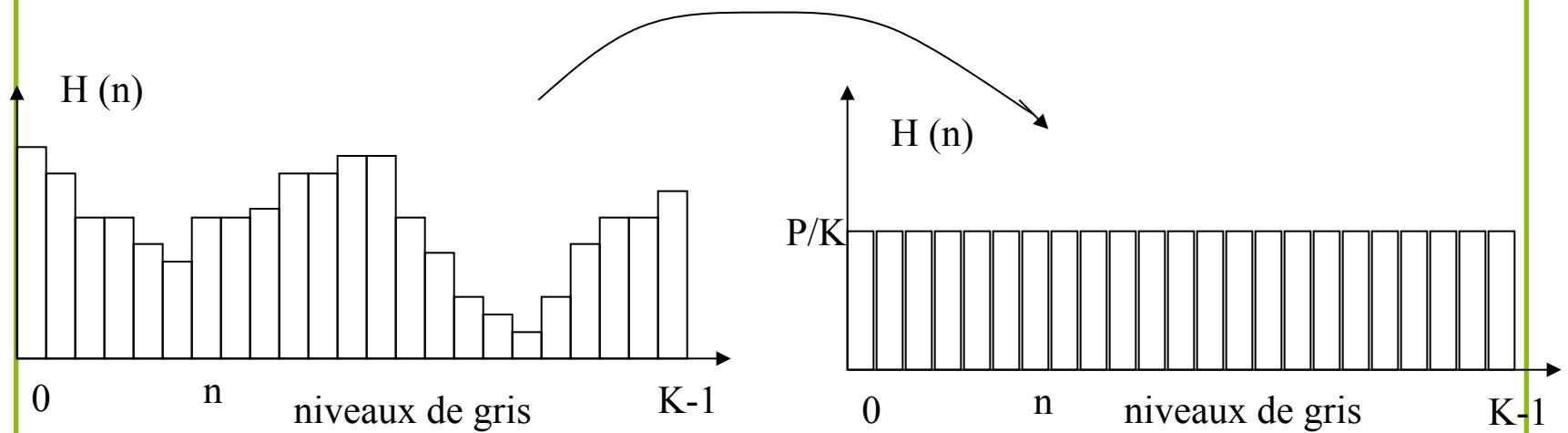


si $L(i, j) \leq S2 \Rightarrow L'(i, j) = 0,$
sinon $L'(i, j) = 255$

- Choix du seuil
 - absolu: seuil fixé pour toute une série d'images
 - adaptatif: déterminé à partir de l'histogramme de chaque image

Normalisation de l'image

- pré-traitements -->
 - permettre la comparaison entre une série d'images,
 - rendre un type de traitement valable sur une série d'images.
- normalisation des données $L'(i, j) = L(i, j) - \bar{m}$ ou $L'(i, j) = \frac{L(i, j) - \bar{m}}{\sigma}$
- égalisation de l'histogramme pour se rapprocher d'un histogramme plat

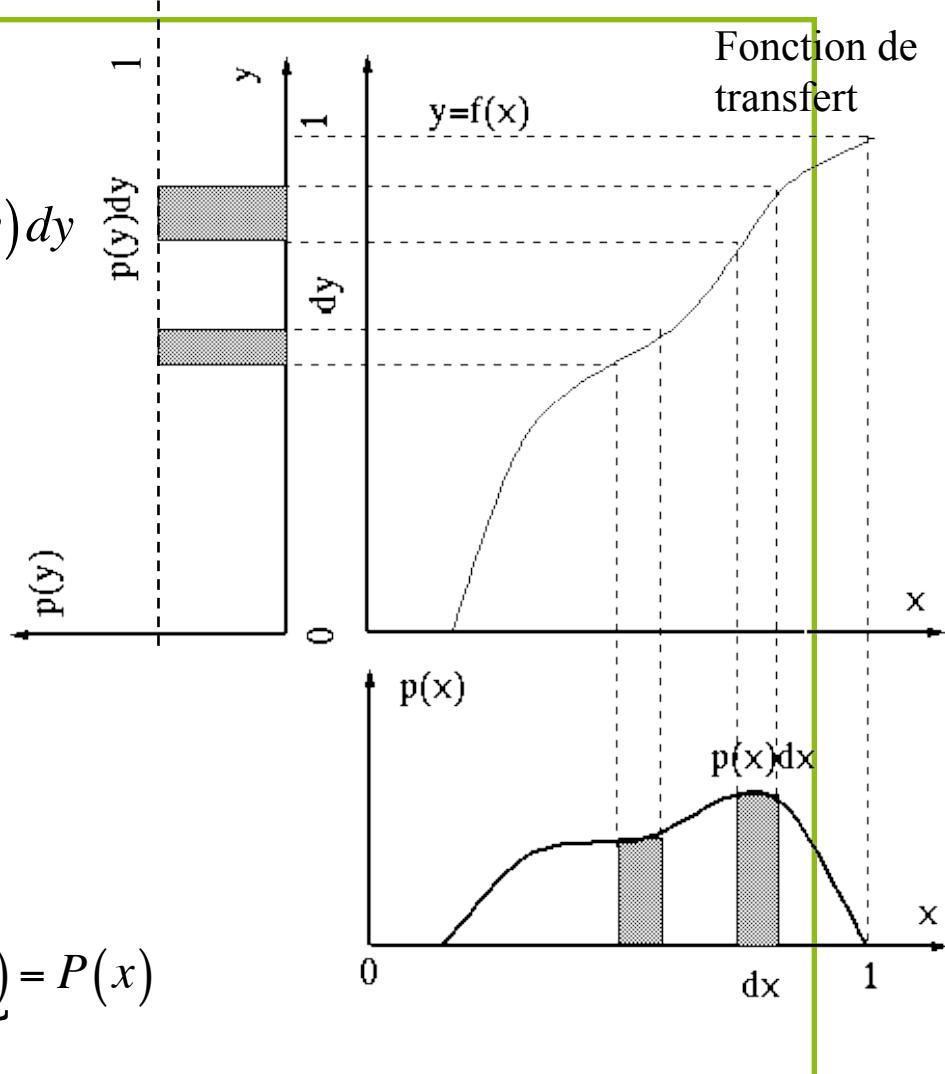


Égalisation d'histogrammes

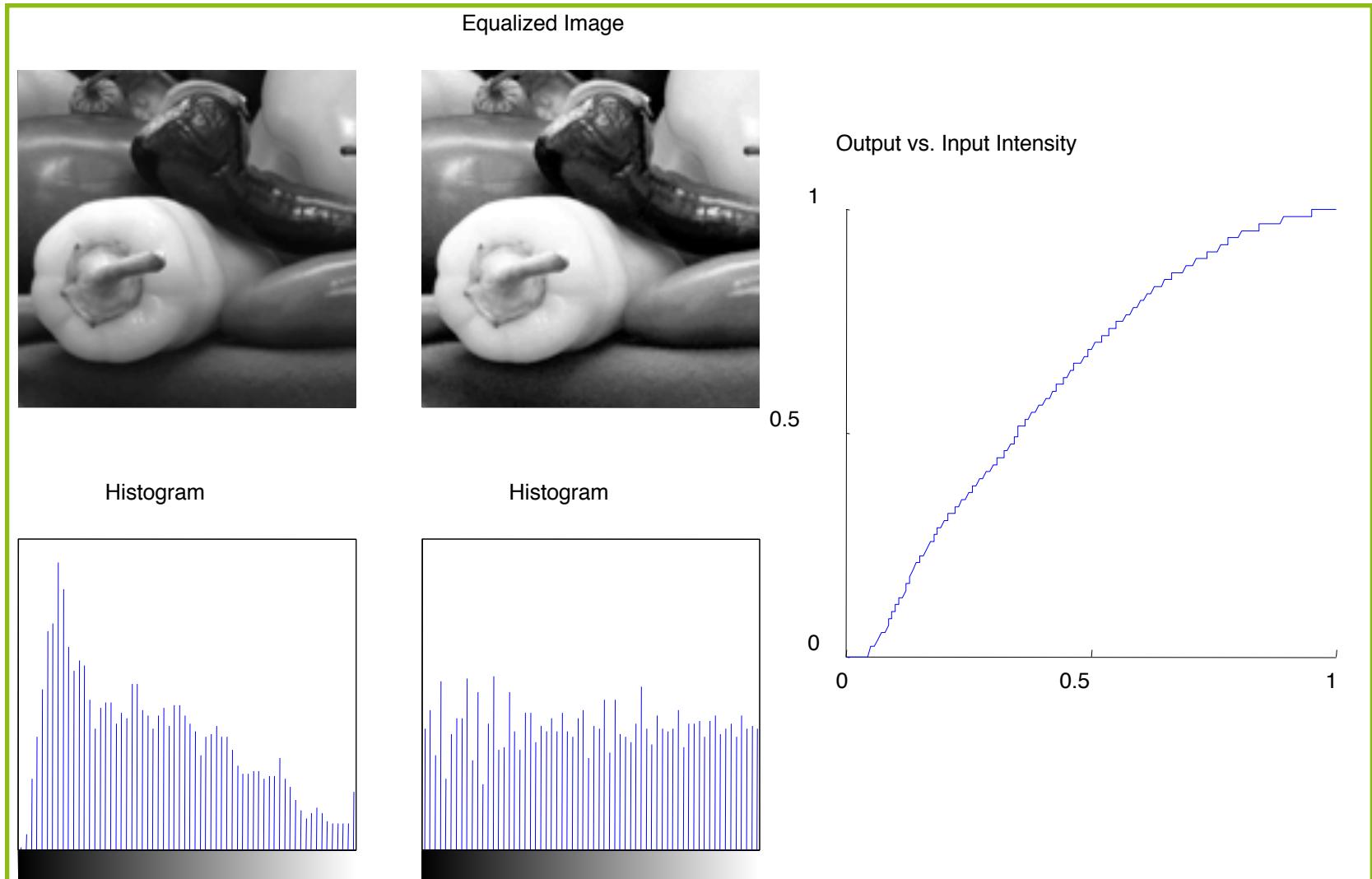
- Histogramme égalisé
 - Qq soit une fonction de transfert (LUT) : $p(x)dx = p(y)dy$
 - Le nombre de pixels transformés par la LUT reste le même
 - But : d'atteindre une densité de probabilité uniforme $p(y) = 1, \forall y$
 $x \in [0 \dots 1], y \in [0 \dots 1]$

$$dy = p(x)dx$$

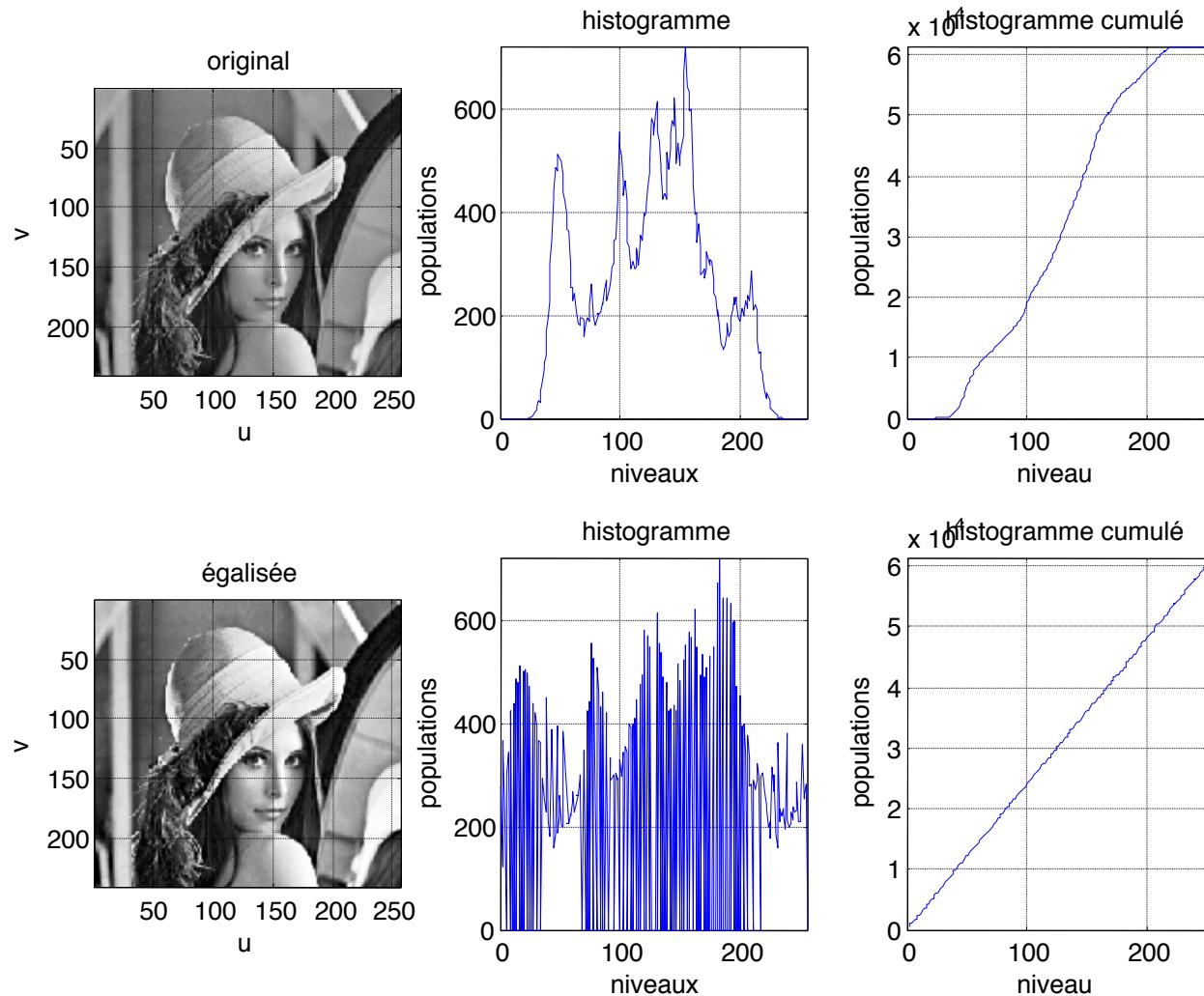
$$y = f(x) = \int_0^x p(t)dt = P(x) - \underbrace{P(0)}_{=0} = P(x)$$



Egalisation d'histogramme



Egalisation d'histogrammes



Amélioration de l'image par modification de l'histogramme (Look Up Tables)

- Changer l'aspect visuel de l'image
- Aucune information nouvelle
- Modification du contraste de l'image
- Recadrage de la dynamique des niveaux de gris de l'image
- LUT ou fonction de transfert : table de transformation des niveaux de gris
- Fausses couleurs en visualisation

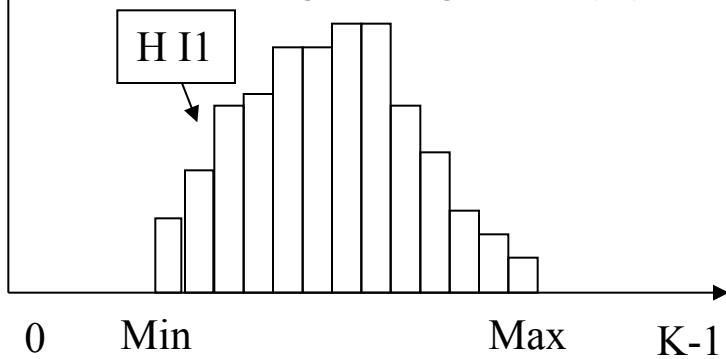
$$I_1(i, j) \xrightarrow{G} I_2(i, j)$$

$G[I_1(i, j)]$ est une application non linéaire,
indépendante de la position du pixel dans l'image
qui s'applique sur les niveaux de gris

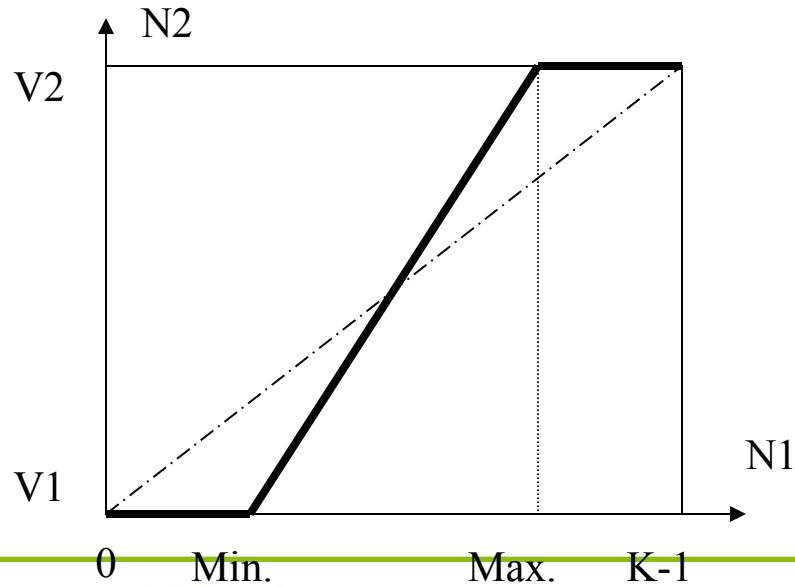
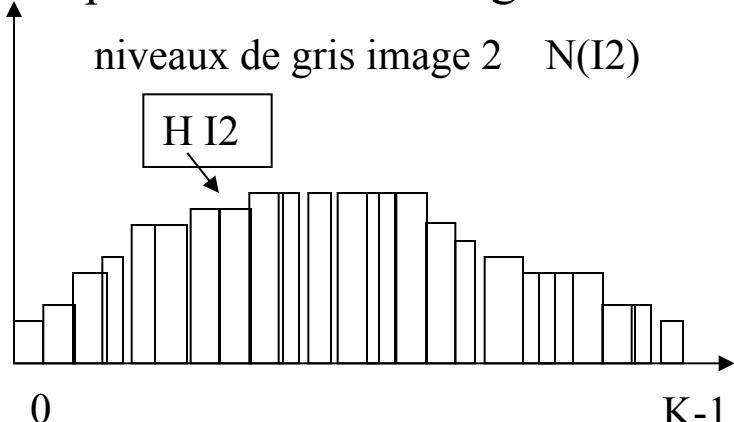
Recadrage de la dynamique de l'image

- Faire occuper à l'image toute la dynamique des niveaux de gris

niveaux de gris image 1 $N(I1)$



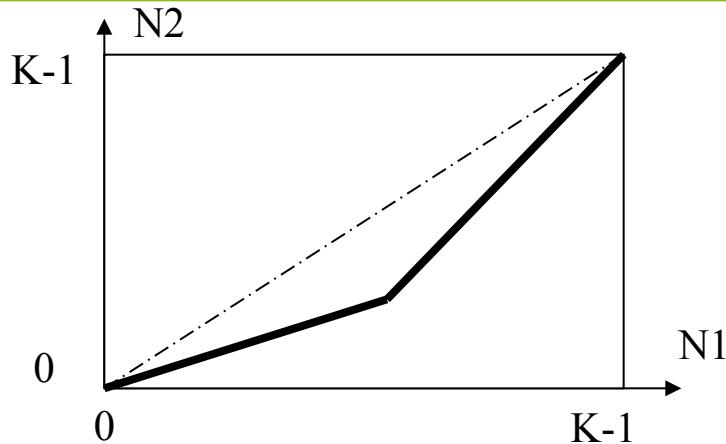
niveaux de gris image 2 $N(I2)$



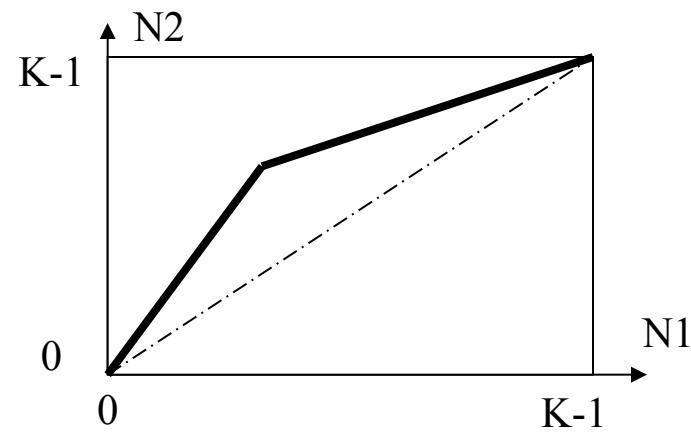
$$\rho = \frac{V2 - V1}{Max - Min}$$

$$N(I2) = Ent[(N(I1) - Min) * \rho] + V1$$

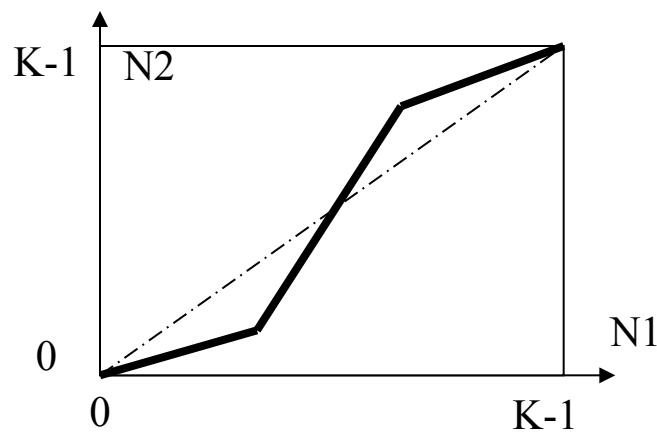
Exemples de LUT



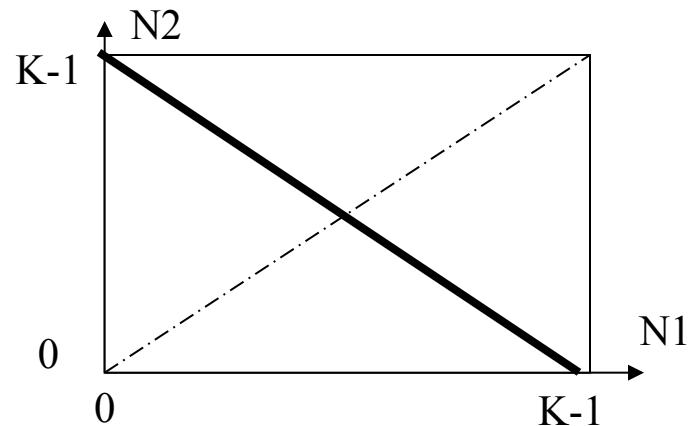
dilatation des zones claires



dilatation des zones sombres



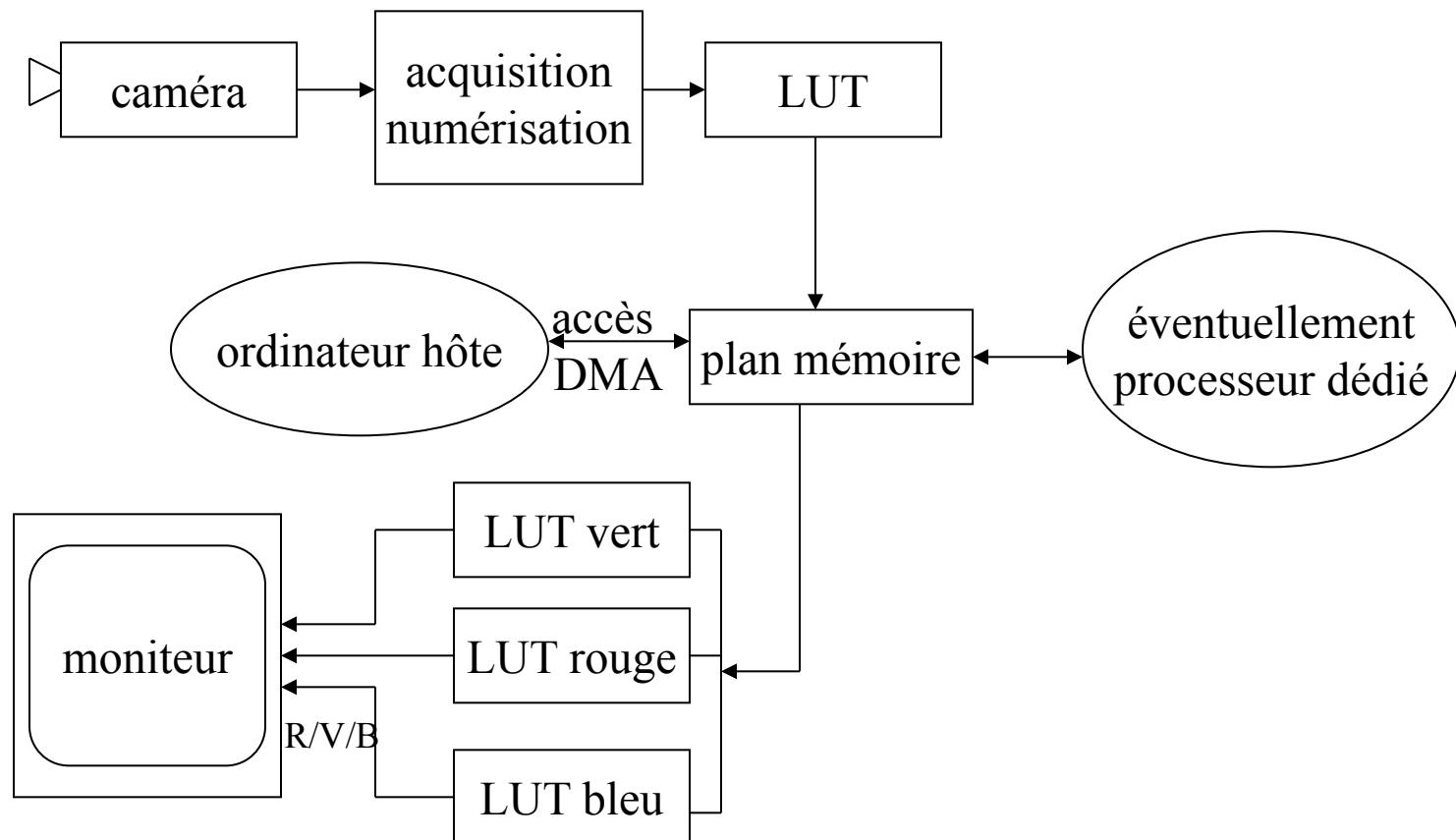
dilatation de la zone milieu



inversion $N \leftrightarrow B$

Acquisition

- Structure d 'une carte ' vision artificielle ' se connectant sur le bus interne d 'un micro-ordinateur hôte

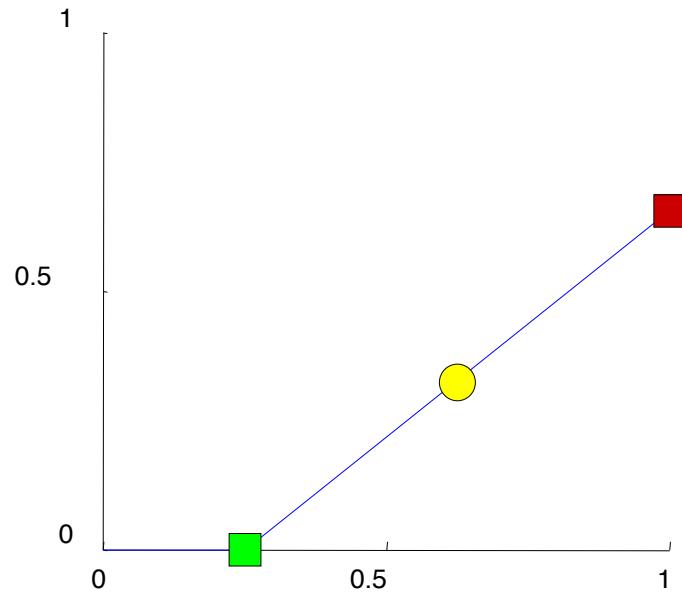


Transformation d'histogramme

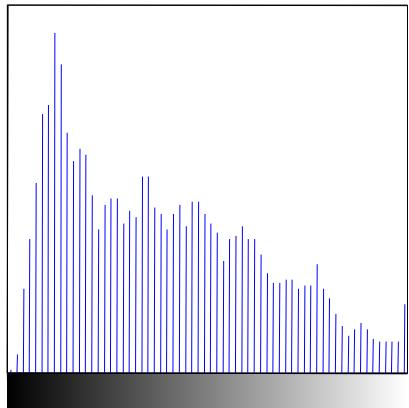
Adjusted Image



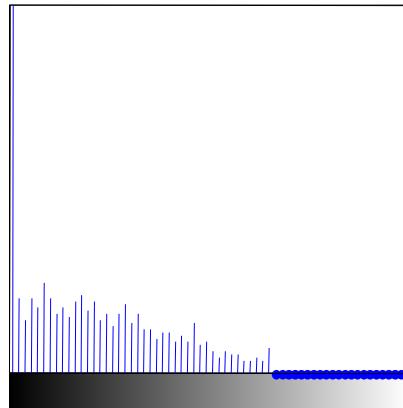
Output vs. Input Intensity



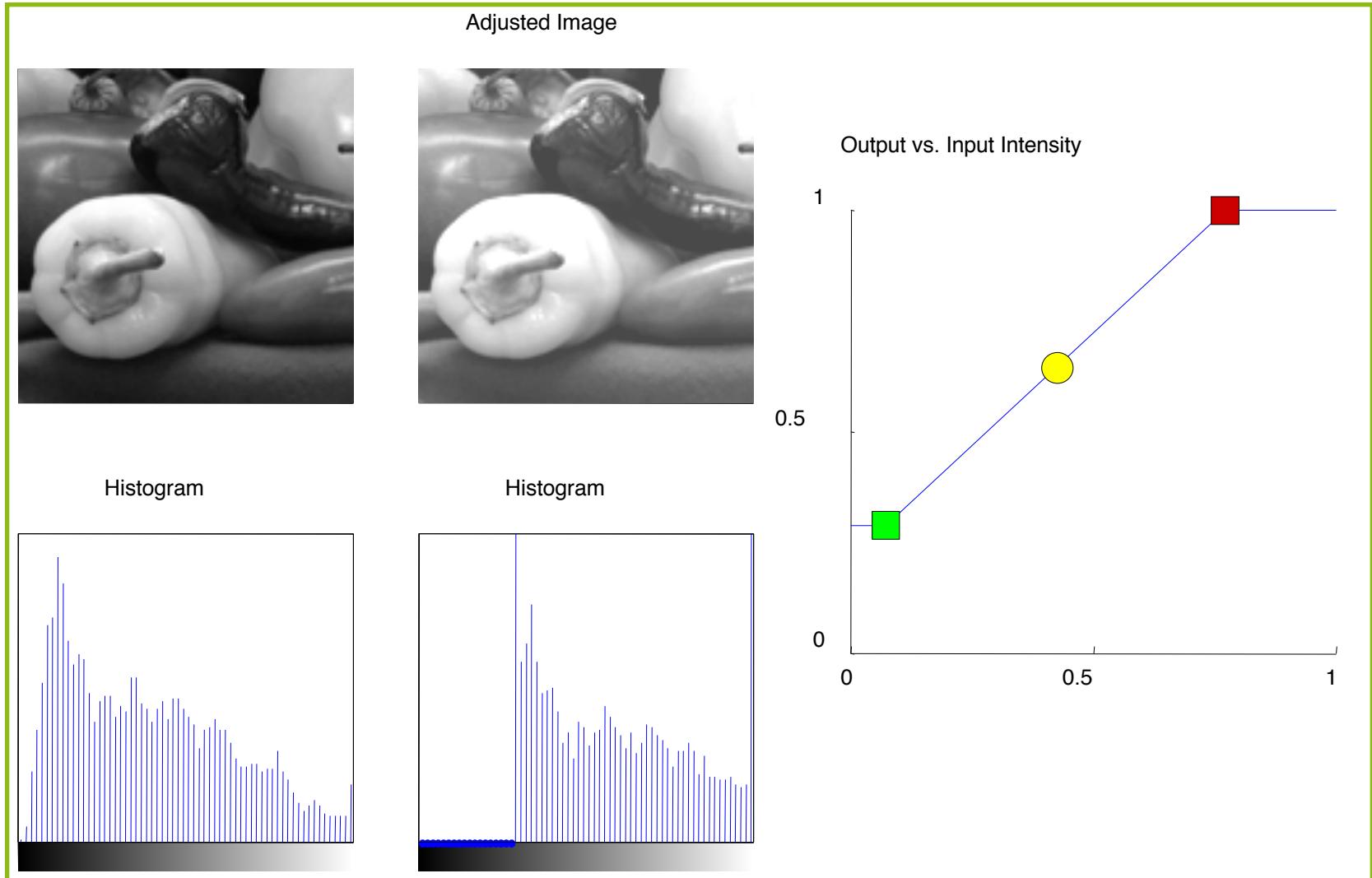
Histogram



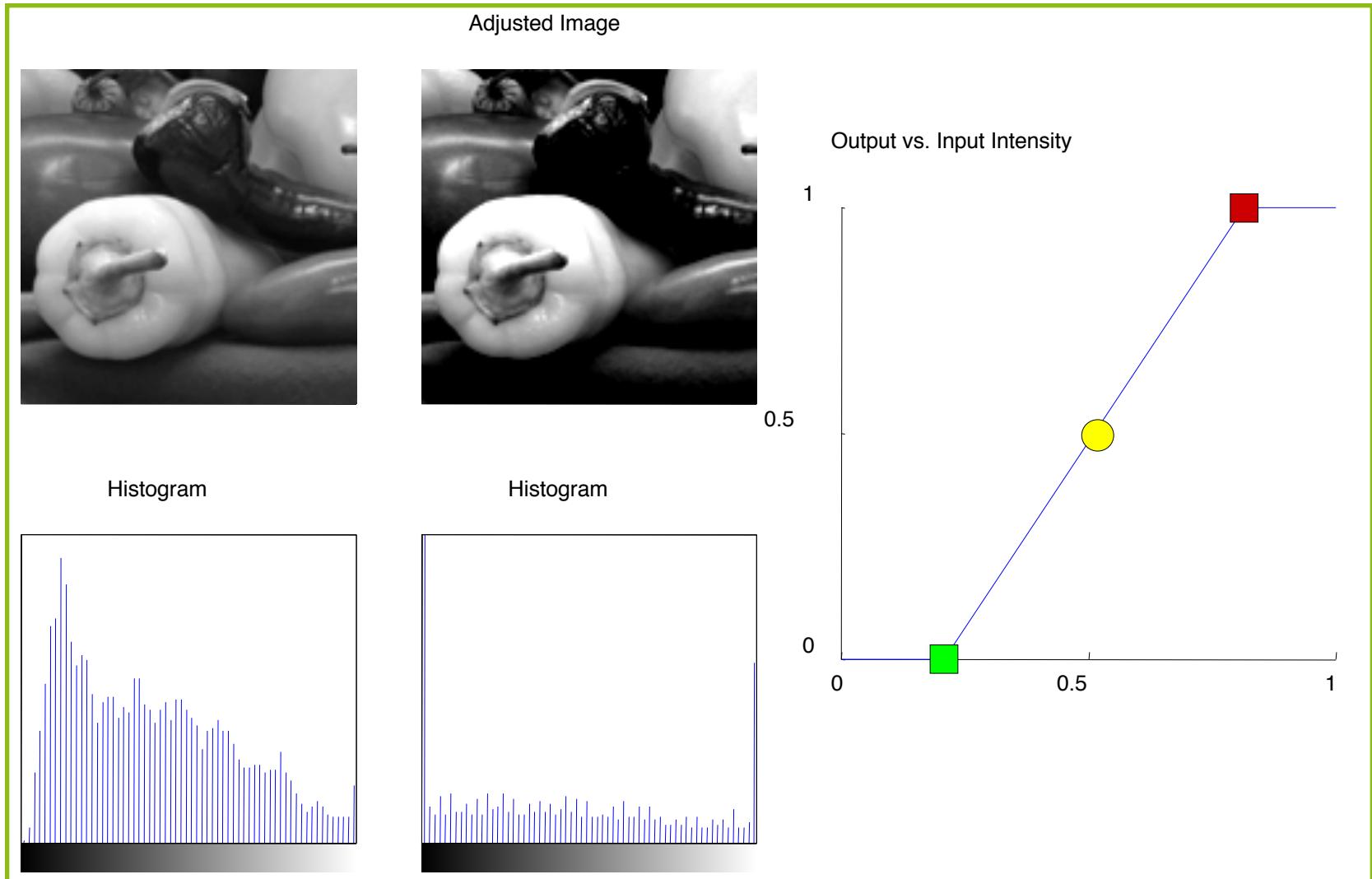
Histogram



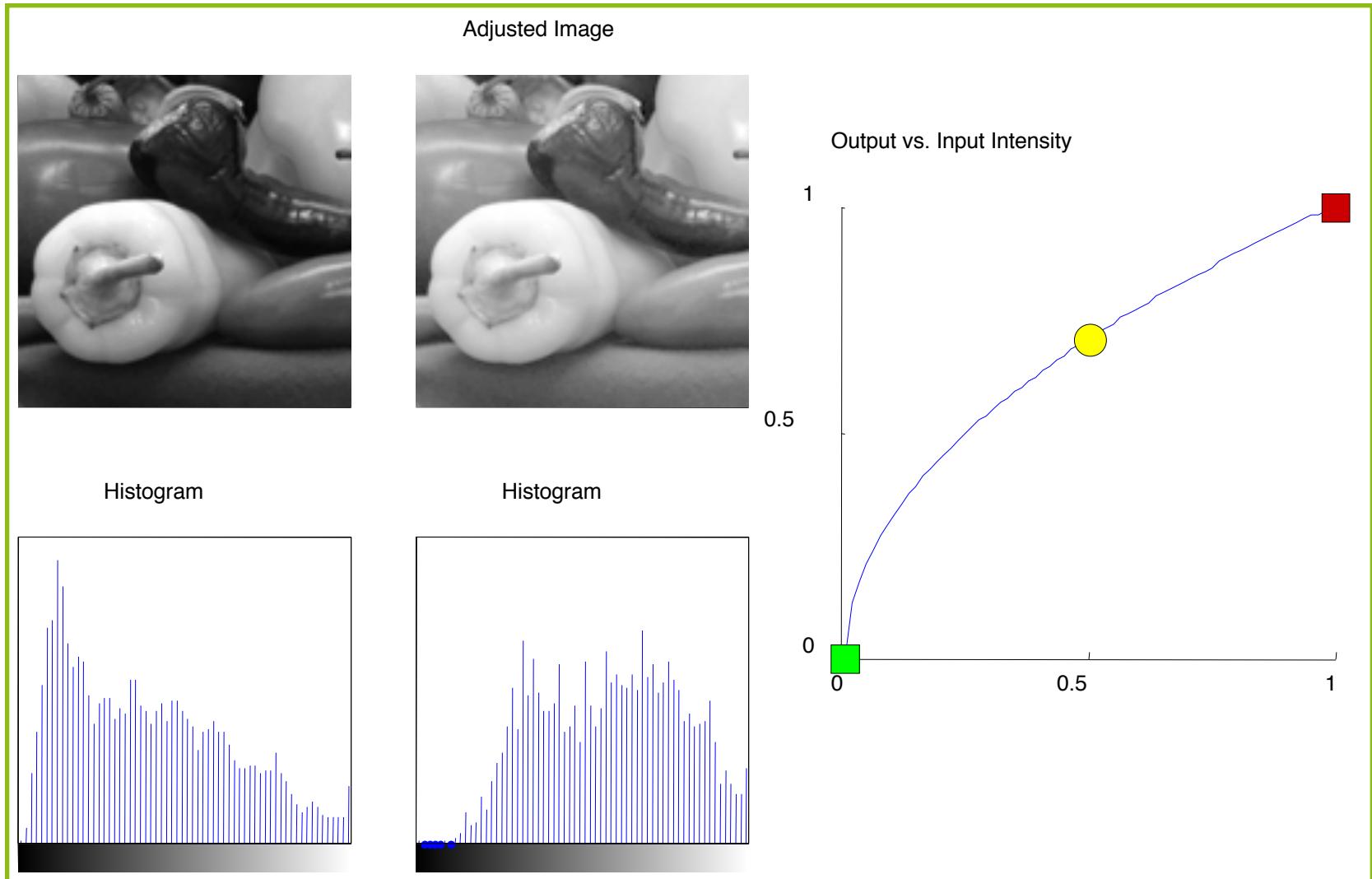
Transformation d'histogramme



Transformation d'histogramme



Transformation d'histogramme



Transformées

- Objectifs:
 - analyse fréquentielle
 - filtrage
 - compression
 - caractérisation de primitives d'intérêt (points particuliers, régions)
 - analyse de texture
- Fourier : décomposition dans une base sinus, cosinus
- Ondelettes : décomposition dans la base des ondelettes choisies (par exemple Haar)
- DCT : décomposition dans une base de cosinus
- Hadamard : décomposition dans la base des fonctions de Walsh
- Hough : Accumulation de paramètres dans un plan (ou espace) de paramètres
 - (ρ, θ) pour les droites
 - $((a,b), R)$ pour les cercles

Transformées

- Généralités sur les transformées
 - Une transformation est un opérateur intégral (dans notre cas discret il s'agit d'une sommation) qui transforme la fonction d'intensité $L(i,j)$ caractéristique de l'image en une fonction $L'(\mu,\nu)$
 - Transformation définie par son noyau $K(i, j, \mu, \nu)$ linéaire définie par :

$$L'(\mu, \nu) = \sum_{i=1}^M \sum_{j=1}^N L(i, j) \cdot K(i, j, \mu, \nu)$$

- On dit que cette transformation est séparable si :

$$K(i, j, \mu, \nu) = K_1(i, \mu) \cdot K_2(j, \nu)$$

Transformées

- Ecriture matricielle des transformées linéaires :

$$L'(\mu, \nu) = \sum_{i=1}^M \sum_{j=1}^N K_1(i, \mu) \cdot L(i, j) \cdot K_2(j, \nu)$$

$$L'_{M \times N} = (K_1)_{M \times N}^T \cdot L_{M \times N} \cdot (K_2)_{M \times N}$$

- Si les matrices sont inversibles et si $M=N$ on aura :

$$L_{N \times N} = (K_2)_{N \times N}^{-1} \cdot L'_{N \times N} \cdot (K_1^T)_{N \times N}^{-1}$$

Transformée de Fourier

- Transformée de Fourier bidimensionnelle

$$X(\mu, \nu) = \sum_{k=-\infty}^{k=+\infty} \sum_{l=-\infty}^{l=+\infty} x(k, l) \cdot e^{-i2\pi(\mu \cdot k + \nu \cdot l)}$$

μ et ν fréquences spatiales

- Propriétés :

- linéaire, séparable, orthogonale, inversible (classiques TF)
- projection sur un espace composé de fonctions sinus et cosinus
- 3 mesures
 - spectre d'amplitude bidimensionnel

$$|X(\mu, \nu)|$$

$$\Theta_X(\mu, \nu) = \arg[X(\mu, \nu)]$$

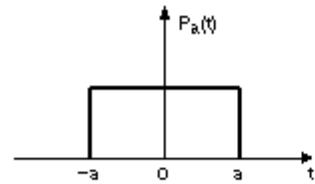
$$|X(\mu, \nu)|^2$$

Propriétés de la Transformée de Fourier

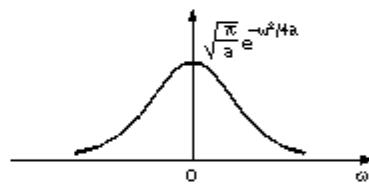
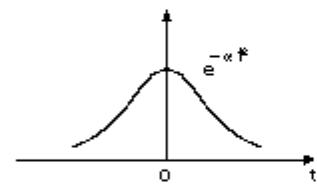
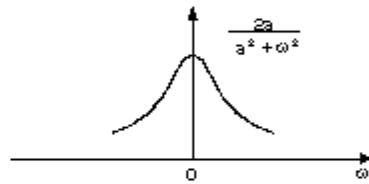
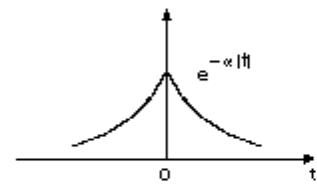
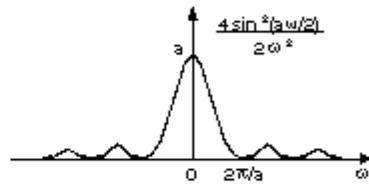
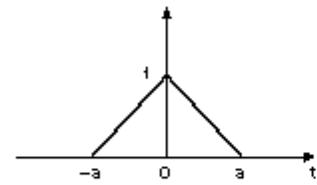
Propriété	Domaine temporel	Domaine fréquentiel
	$f(t)$	$F(\omega)$ avec $\omega = 2\pi\nu$
Linéarité	$af_1(t) + bf_2(t)$	$aF_1(\omega) + bF_2(\omega)$
Convolution	$f_1(t) \otimes f_2(t)$	$F_1(\omega) \cdot F_2(\omega)$
Conjugaison complexe	$\overline{f(t)}$	$\overline{F(-\omega)}$
Echelle temporelle	$f(at)$	$\frac{1}{ a }F\left(\frac{\omega}{a}\right)$
Inversion temporelle	$f(-t)$	$F(-\omega)$
Décalage temporel	$f(t - t_0)$	$e^{-i\omega t_0}F(\omega)$
Modulation cos	$f(t) \sin \omega_0 t$	$\frac{i}{2}[F(\omega + \omega_0) - F(\omega - \omega_0)]$
Modulation sin	$f(t) \cos \omega_0 t$	$\frac{1}{2}[F(\omega + \omega_0) + F(\omega - \omega_0)]$
Décalage fréquentiel	$f(t)e^{i\omega_0 t}$	$F(\omega - \omega_0)$
Intégration	$\int_{-\infty}^t f(u)du$	$\frac{1}{i\omega}F(\omega) + \pi F(0)\delta(\omega)$
Th. de Parseval	$\int_{\mathbb{R}} f(t) \cdot \overline{g(t)} dt = \int_{\mathbb{R}} F(\omega) \cdot \overline{G(\omega)} d\omega$	

Transformées de Fourier classiques

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} d\omega$$

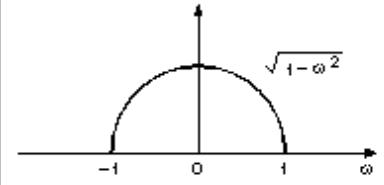
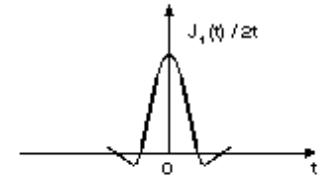
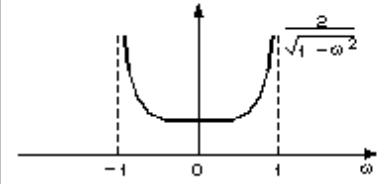
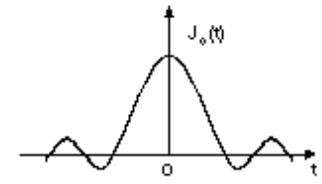
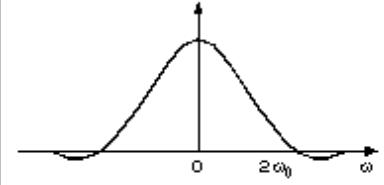
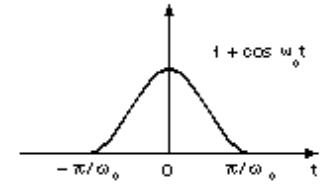


$$F(t) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$$



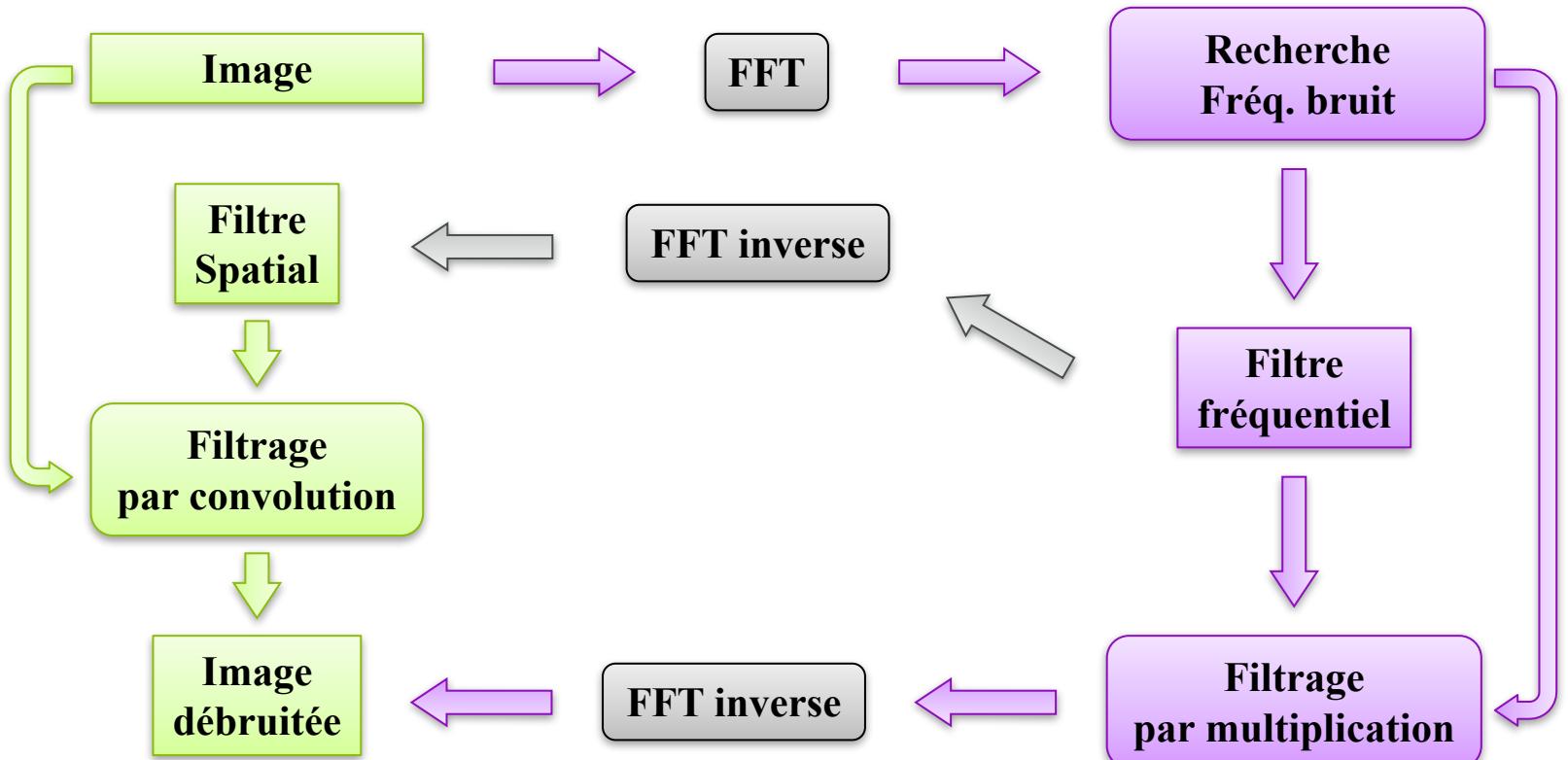
$$\delta(t) \stackrel{F}{\Leftrightarrow} 1$$

$$1 \stackrel{F}{\Leftrightarrow} \sqrt{2\pi} \delta(\omega)$$

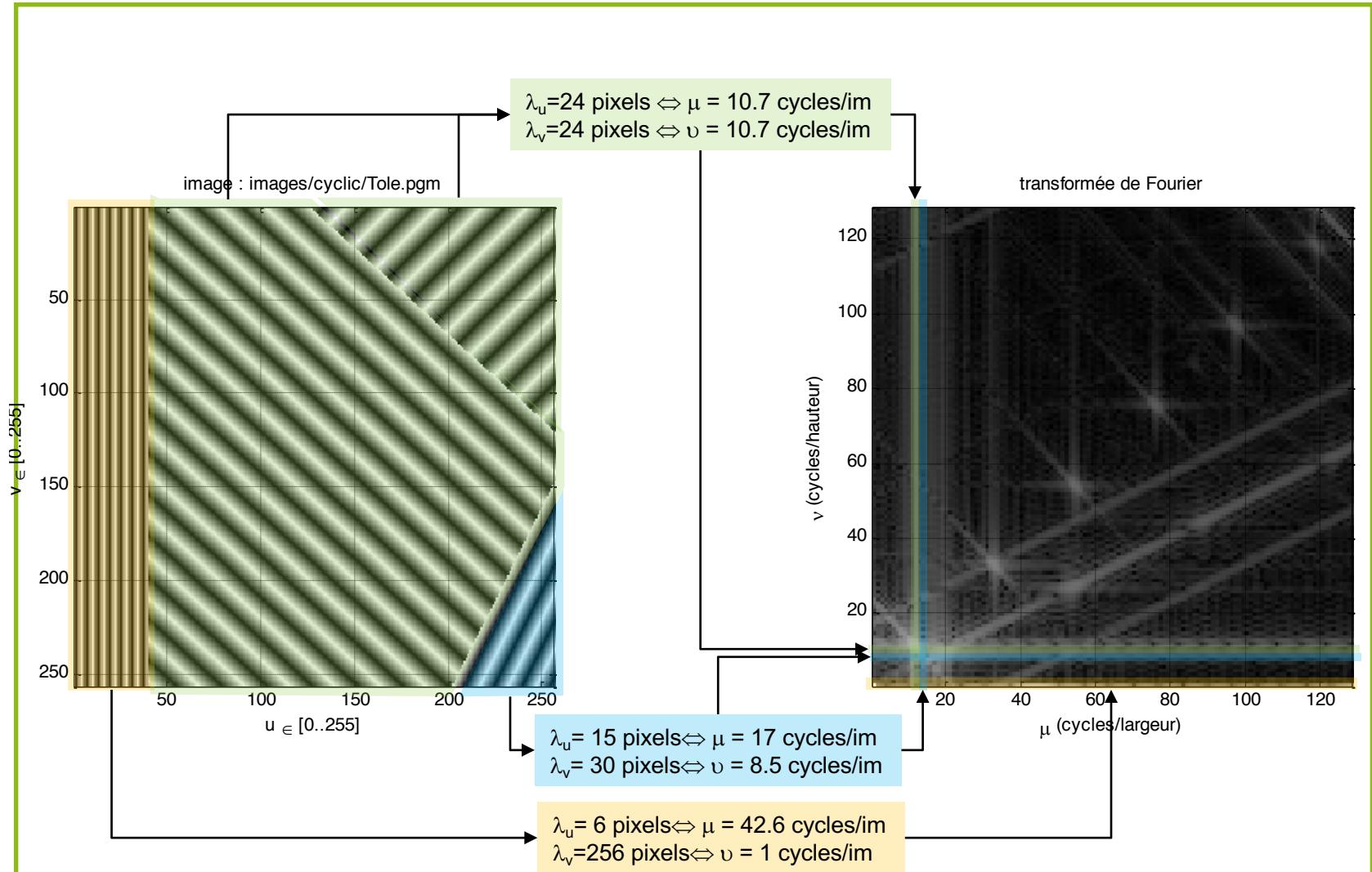


Transformée de Fourier et filtrage

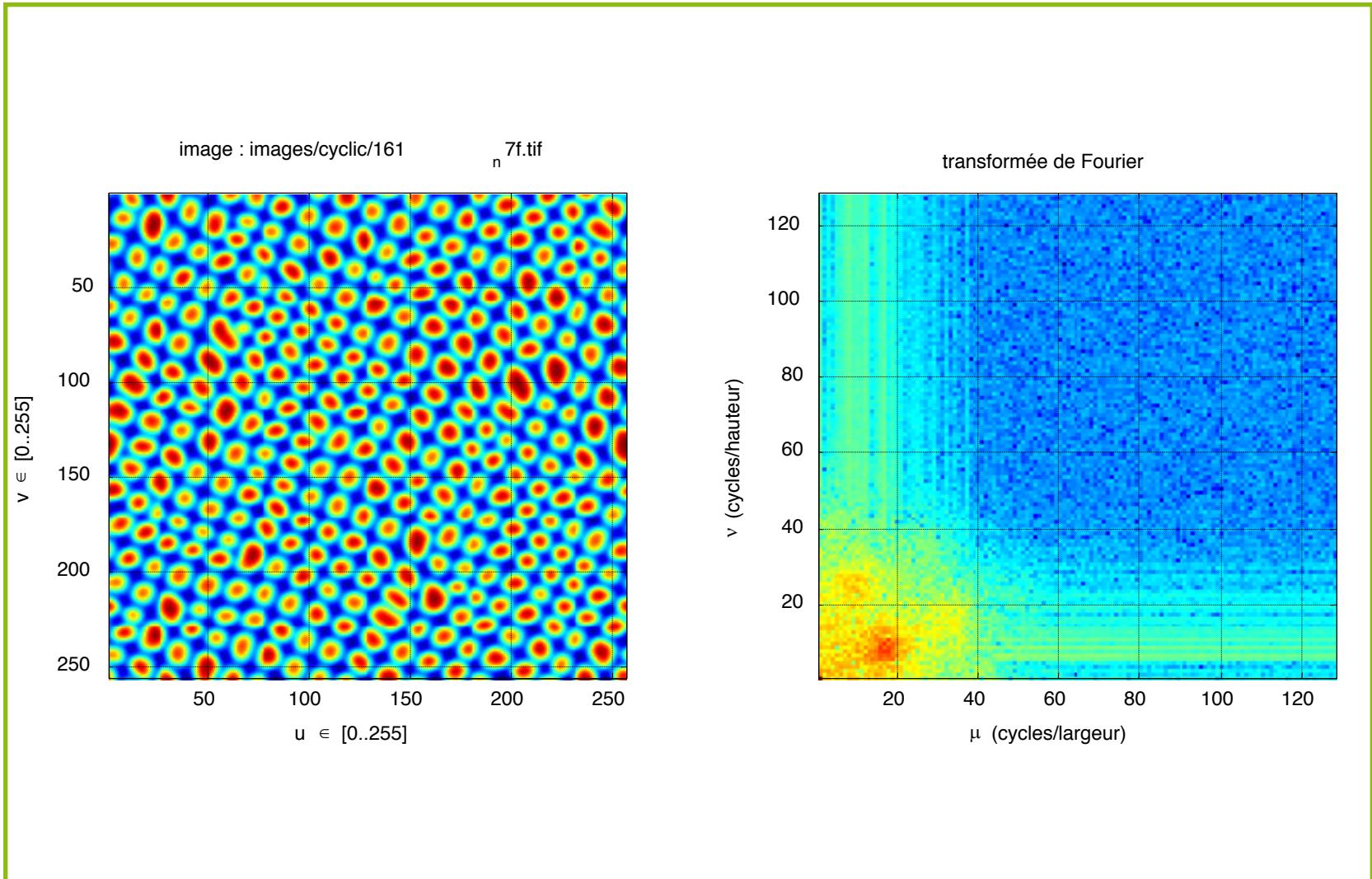
- Exemple de filtrage fréquentiel :
 - élimination d'un bruit de fréquence connue



Transformée de Fourier : exemple 1

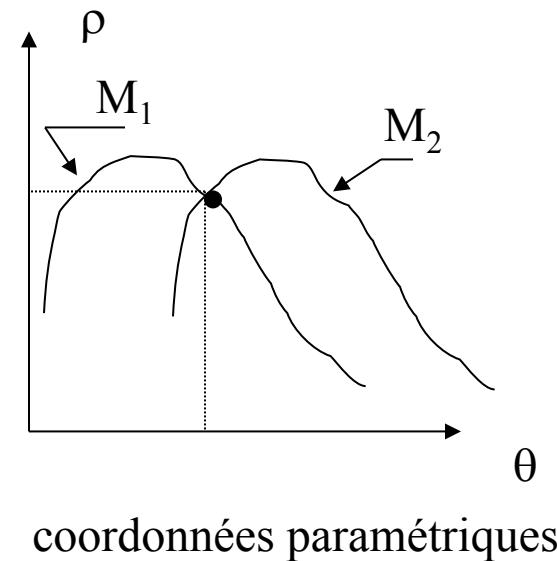
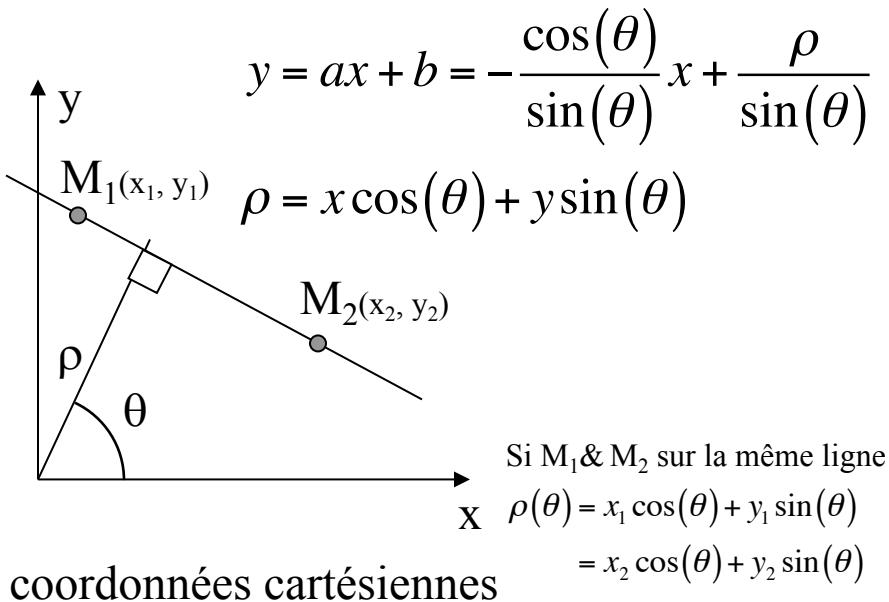


Transformée de Fourier: exemple 2



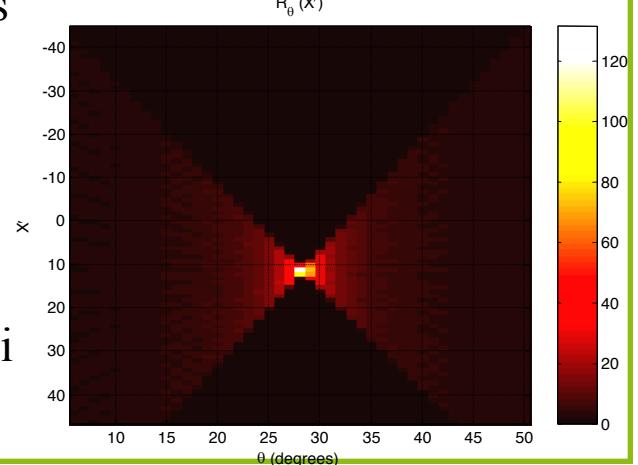
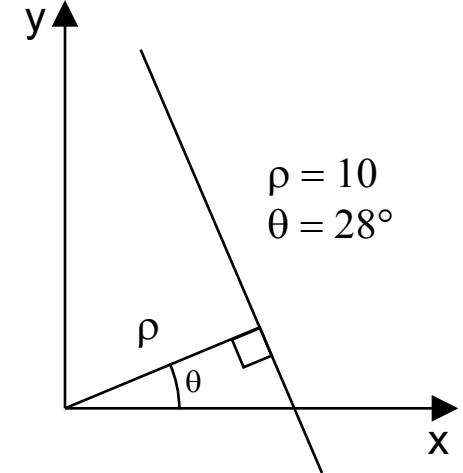
Transformée de Hough

- algorithme d'accumulation de points dans l'espace des paramètres de la primitive recherchée
- transposition coordonnées cartésiennes → coordonnées paramétriques
 - Paramètres détection de lignes : ρ & θ (plan d'accumulation)



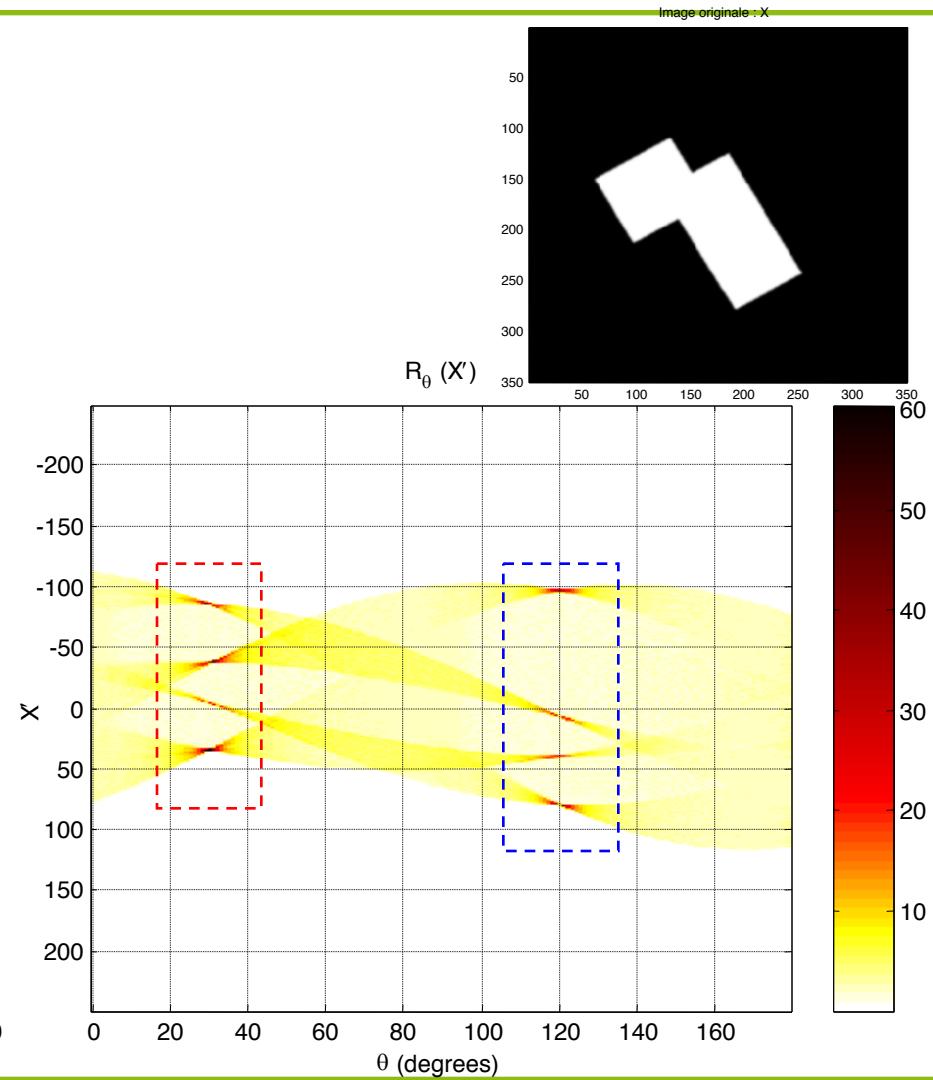
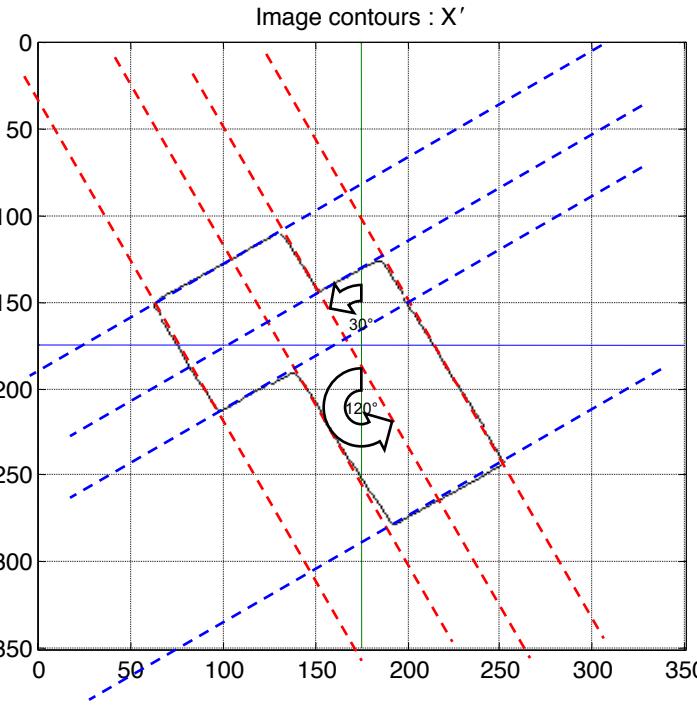
Transformée de Hough

- Segment de droite : $x\cos(\theta) + y\sin(\theta) = \rho$
 - ρ : distance normale à l'origine
 - θ : angle de la droite
- Principe
 - transposition coordonnées cartésiennes
→ coordonnées paramétriques
 - algorithme d'accumulation de points dans l'espace des paramètres
 - Appliqué aux points (x_i, y_i) des contours de l'image
- Applications
 - Détection de lignes
 - Détection d'autres formes
 - Dépends des paramètres de l'espace (ici un plan) d'accumulation



Transformée de Hough (exemple 1)

- Image originale



Transformée de Hough (exemple 2)

- Image originale

Image originale : X

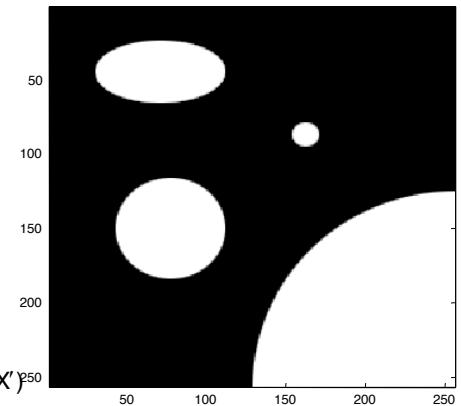
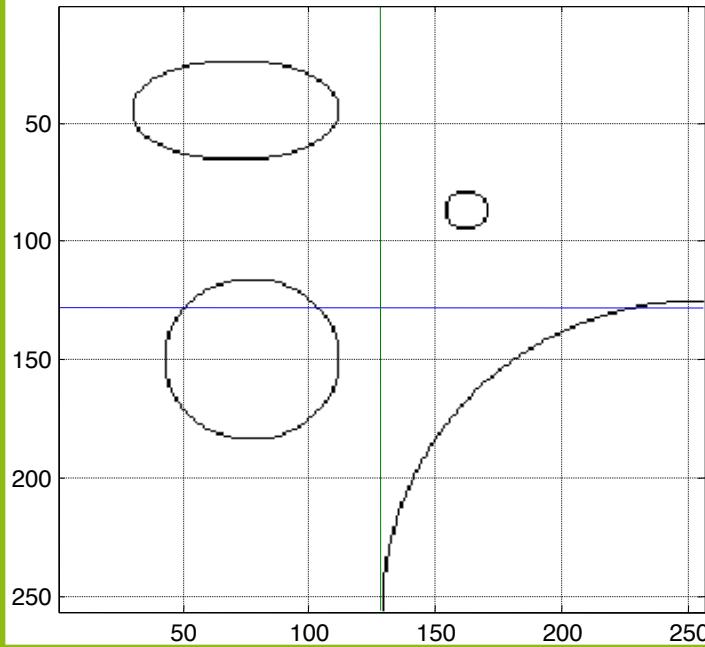
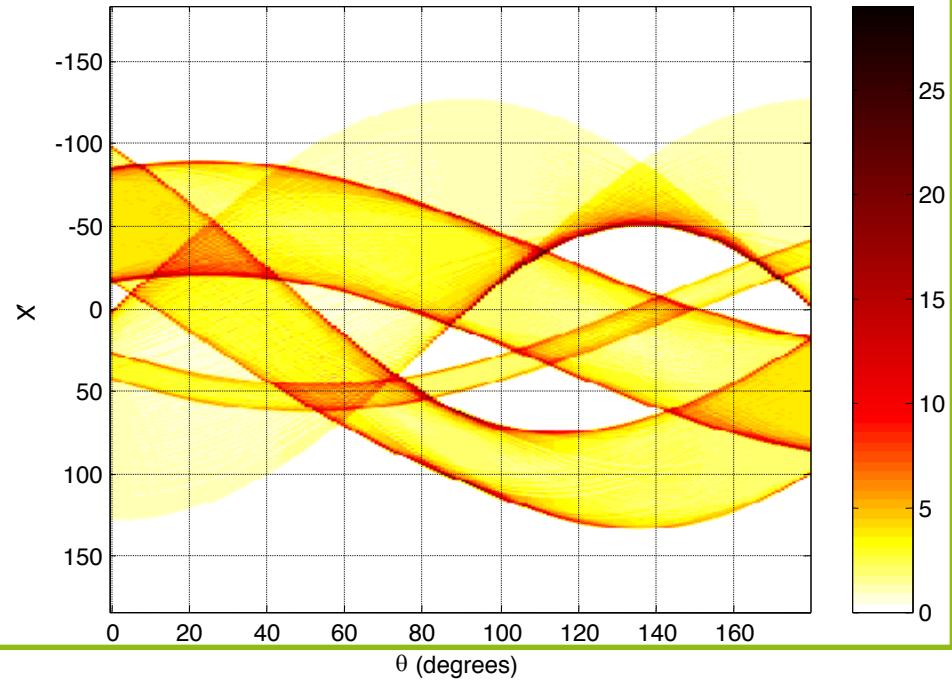


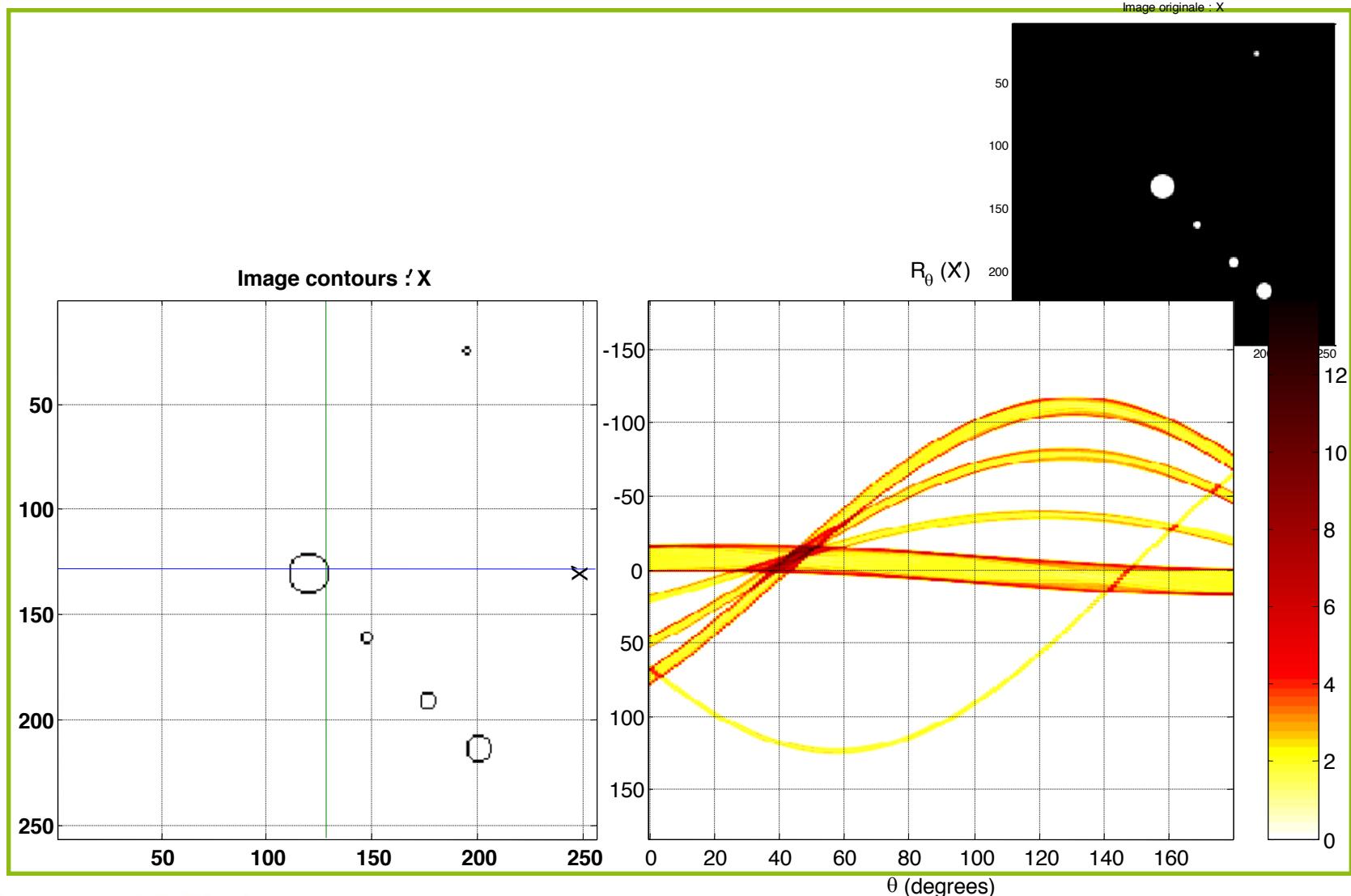
Image contours : X'



$R_\theta (X')$



Transformée de Hough





Filtrage & Segmentation

Traitement et Filtrage des images

- Objectifs:
 - amélioration de la qualité d'une image
 - recherche de primitives d'intérêt : contours, points, etc.
 - restauration d'image
- Filtrage linéaire
 - opérateurs: $+$, \times , Σ , structure: non récursif ou récursif
- Filtrage non linéaire
 - comporte au moins un opérateur logique
- Traitements structurants
 - morphologie mathématique

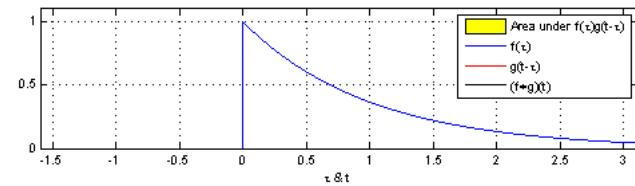
Filtrage monodimensionnel

- Filtrage discret
 - Combinaison linéaire des échantillons du signal avec les coefficients du filtre \Rightarrow Convolution

$$y = x \otimes f$$

continu

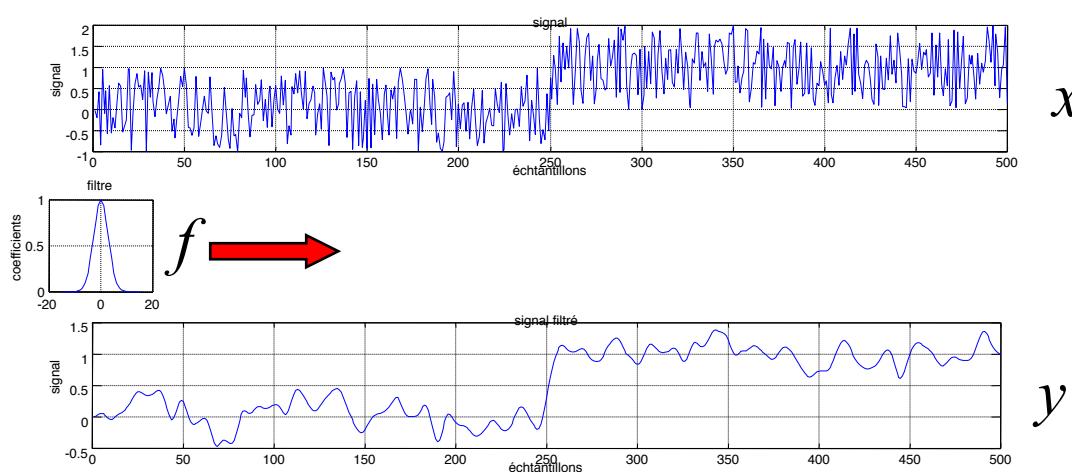
$$y(t) = \int_{-\infty}^{+\infty} x(\tau) f(t - \tau) d\tau = \int_{-\infty}^{+\infty} x(t - \tau) f(\tau) d\tau$$



discret

$$y(i) = \sum_{i'=-K/2}^{i'=K/2} f(i - i') x(i')$$

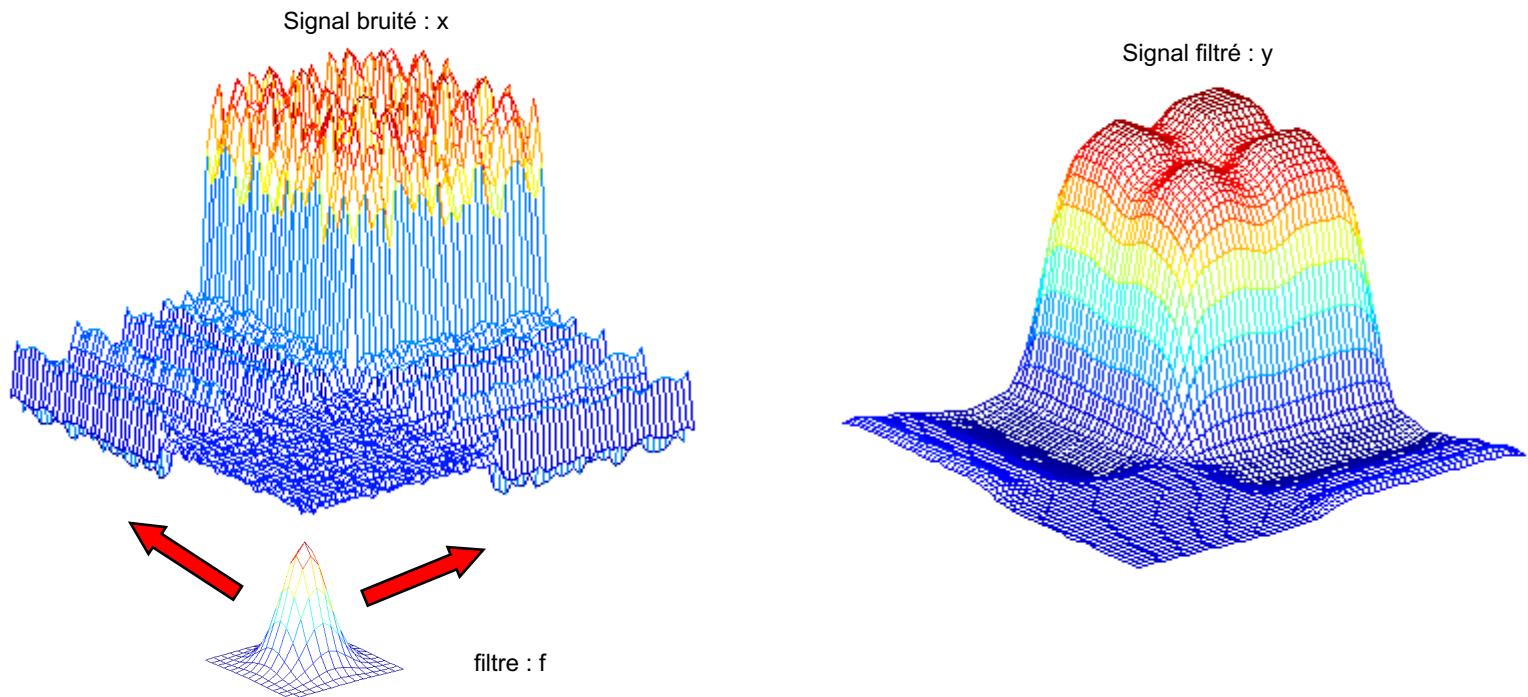
$$= \sum_{i'=-K/2}^{i'=K/2} f(i') x(i - i')$$



Filtrage bidimensionnel

- Filtrage bi-linéaire

$$y(k,l) = \sum_{k'=-K/2}^{K/2} \sum_{l'=-L/2}^{L/2} f(k',l') x(k-k',l-l')$$



Filtrage linéaire

- Produit de convolution bi-dimensionnel entre deux signaux $x(k,l)$ et $h(k,l)$

$$y(k,l) = \sum_{k'=-\infty}^{k'=+\infty} \sum_{l'=-\infty}^{l'=+\infty} x(k - k', l - l') h(k', l')$$

- $x(k,l)$ représente une image comprenant M lignes et N colonnes et $h(k,l)$ la réponse impulsionale du filtre contenant M_2 lignes et N_2 colonnes (en général $M_2 \ll M$ et $N_2 \ll N$)

Filtrage passe-bas

- Filtrage passe bas
 - élimination d'un bruit 'haute fréquence' décorrélaté de l'image
 - lissage de l'image (filtre moyenneur) : adoucissement des contours
 - Filtre statistique (non linéaire)
- Exemple de filtrage passe bas statistique
 - Pour chaque pixel centré sur X

	O1	O2	O3	
	O4	X	O5	
	O6	O7	O8	

si $\left| X - \frac{1}{8} \sum_{i=1}^8 O_i \right| > Seuil$

alors $X = \frac{1}{8} \sum_{i=1}^8 O_i$

sinon $X = X$

Filtrage passe-bas linéaire

- Convolution par un masque prédéfini
 - Masque 3x3 appliqué sur l'image

$$L'(i,j) = \sum_{u=0}^2 \sum_{v=0}^2 L(i-u+1, j-v+1) \cdot h(u,v)$$

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$\underbrace{}_{h_1(u,v)}$

$$\frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$\underbrace{}_{h_2(u,v)}$

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$\underbrace{}_{h_3(u,v)}$

- Gain statique du masque

$$G = \sum_{u=0}^2 \sum_{v=0}^2 h(u,v) = 1$$

Filtrage passe-haut

- Filtrage passe-haut
 - en utilisant la différence avec un filtrage passe bas
 - élimination ou atténuation des fréquences spatiales basses : accentuation des contours
 - création d'une image masque

$$L_M(i, j) = c \cdot \underbrace{L(i, j)}_{\substack{\textit{image} \\ \textit{initiale}}} - (1 - c) \cdot \underbrace{L_{PB}(i, j)}_{\substack{\textit{image} \\ \textit{filtrée(PB)}}}$$

$$\frac{3}{5} \leq c \leq \frac{5}{6}$$

Si PB = gaussienne

Filtrage passe-haut (suite)

- Filtrage statistique (non linéaire)
 - pour un voisinage de P points

$$L'(i, j) = L(i, j) \cdot \sigma_P(i, j)$$

$$\sigma_P^2(i, j) = \frac{1}{P} \sum_i \sum_j [L(i, j) - \bar{L}(i, j)]^2$$



Filtrage passe-haut

- Convolution par un masque (linéaire)
 - filtre appliqué sur l'image

$$L'(i, j) = \sum_{u=0}^2 \sum_{v=0}^2 L(i - u + 1, j - v + 1) \cdot h(u, v)$$

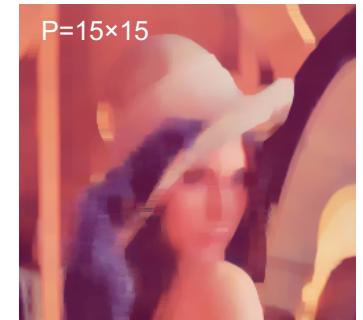
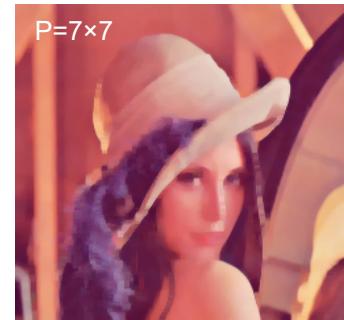
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \underbrace{\qquad\qquad\qquad}_{h_1(u,v)}$$

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \underbrace{\qquad\qquad\qquad}_{h_2(u,v)}$$

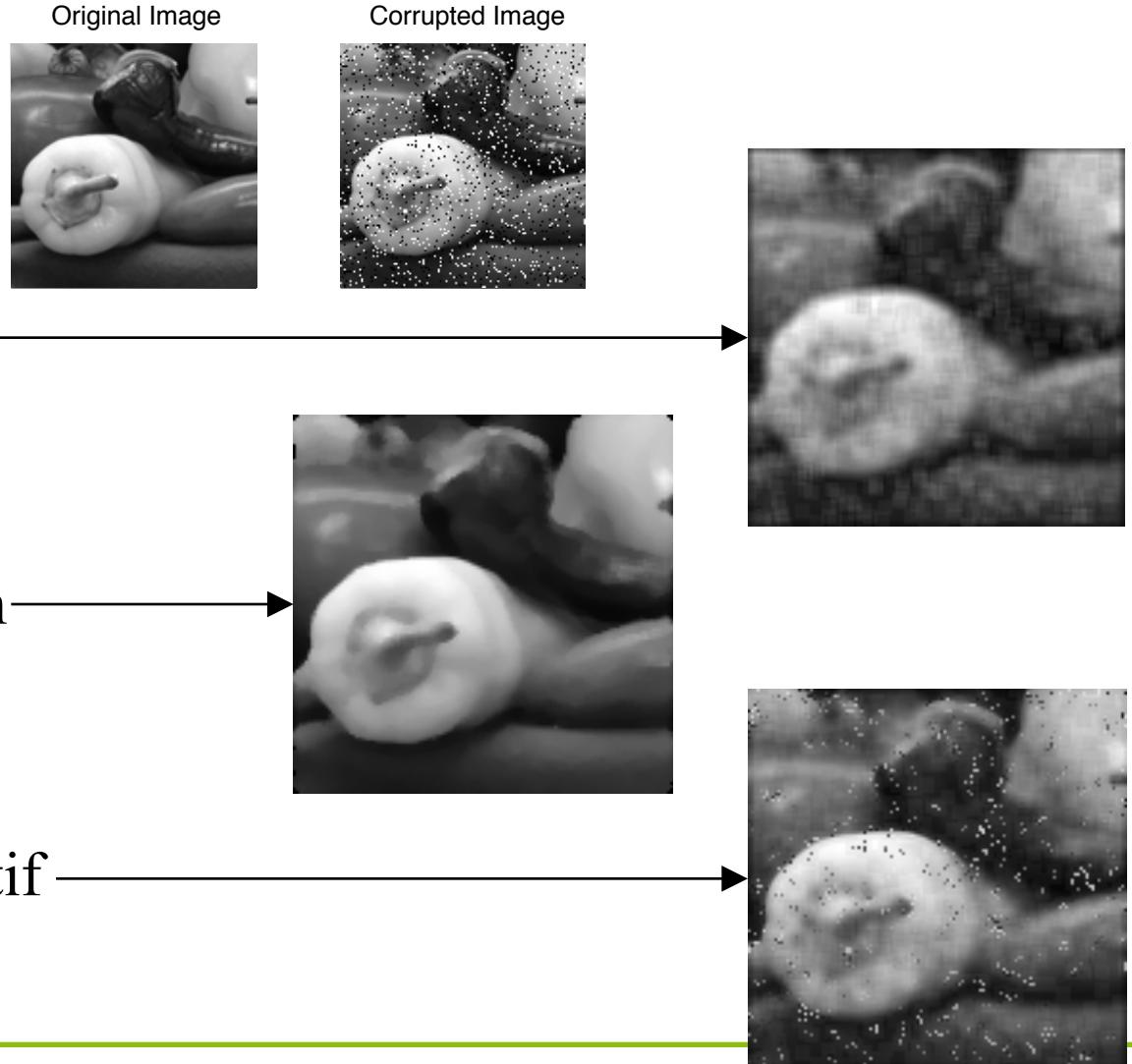
$$\begin{bmatrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{bmatrix} \underbrace{\qquad\qquad\qquad}_{h_3(u,v)}$$

Filtrage

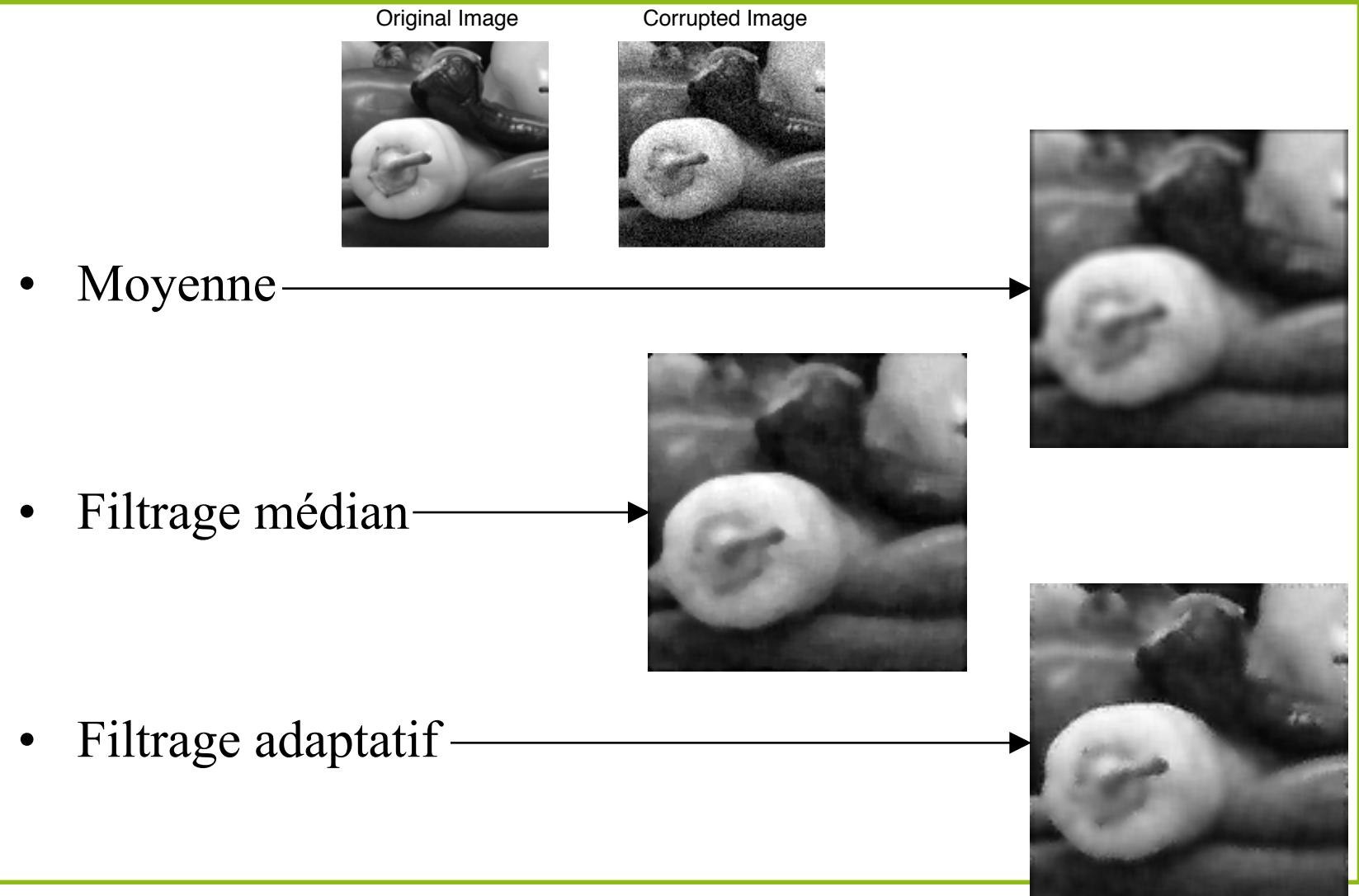
- Filtre médian
 - opérateur non linéaire
 - efficace pour supprimer un bruit dû à des impulsions parasites
 - exemple en mono dimensionnel
 - on extrait une fenêtre de $2n+1$ valeurs consécutives
 - 12 | 6 | 8 | 2 | 15 | 21 | 10 | 18 | 34
 - on les classe par ordre croissant
 - 2 | 6 | 8 | 10 | 12 | 15 | 18 | 21 | 34
 - la médiane est la valeur centrale : 12
 - cette valeur remplacera 15 dans la fenêtre précédente



Filtrage des images : Filtres 5x5, bruit poivre & sel



Filtrage des images : Filtres 5x5, bruit gaussien

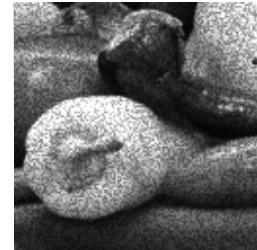


Filtrage des images : Filtres 5x5, bruit Speckle

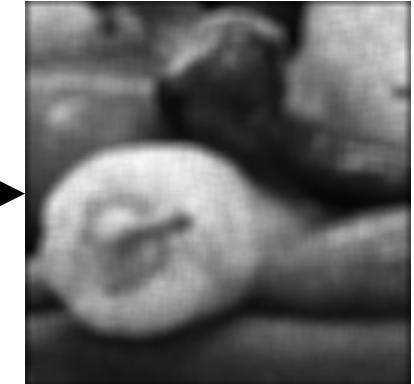
Original Image



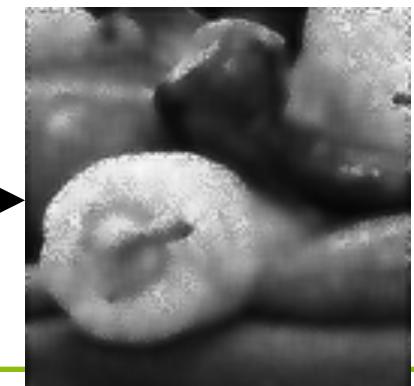
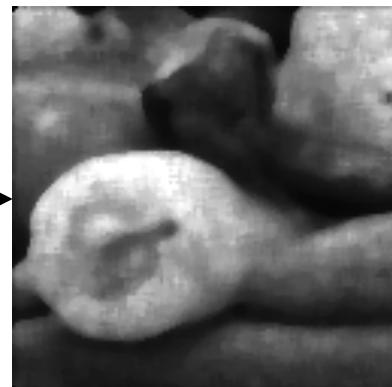
Corrupted Image



- Moyenne



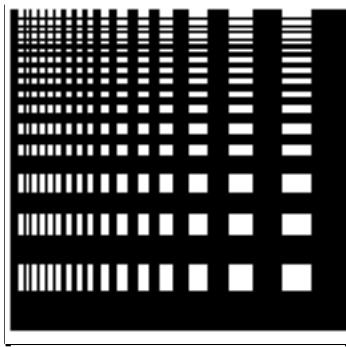
- Filtrage médian



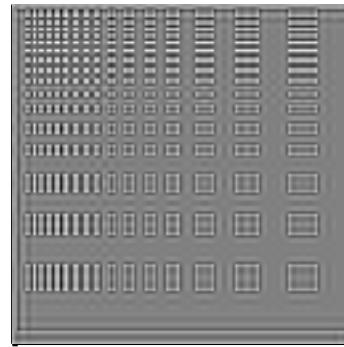
Filtrage des images

- Filtrage passe-haut : taille 15x15, fréquence de coupure $\frac{1}{2}$ cycle/pixel

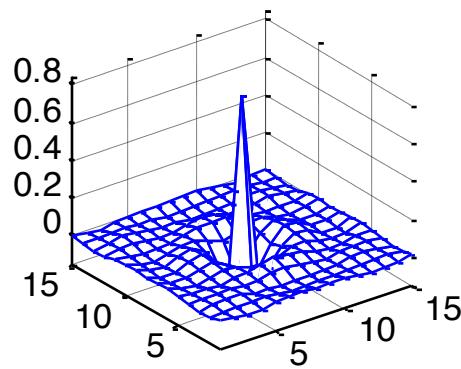
Original Rectangles Image



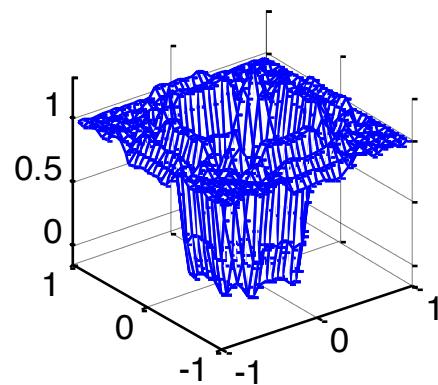
Filtered Image



Filter Coefficients



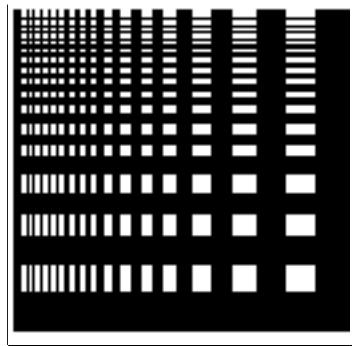
Frequency Response



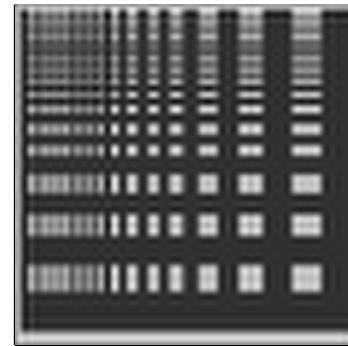
Filtrage des images

- Filtrage passe-bas : taille 15x15, fréquence de coupure $\frac{1}{2}$ cycle/pixel

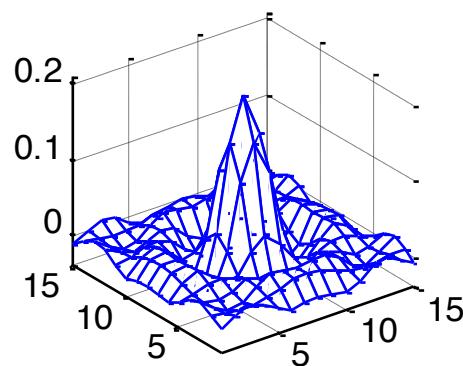
Original Rectangles Image



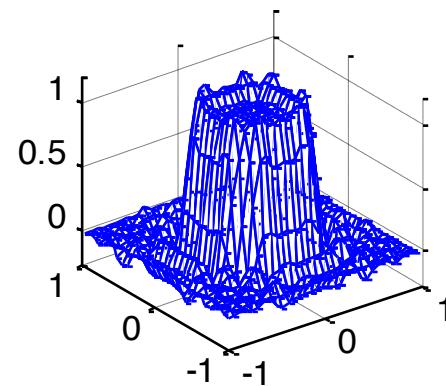
Filtered Image



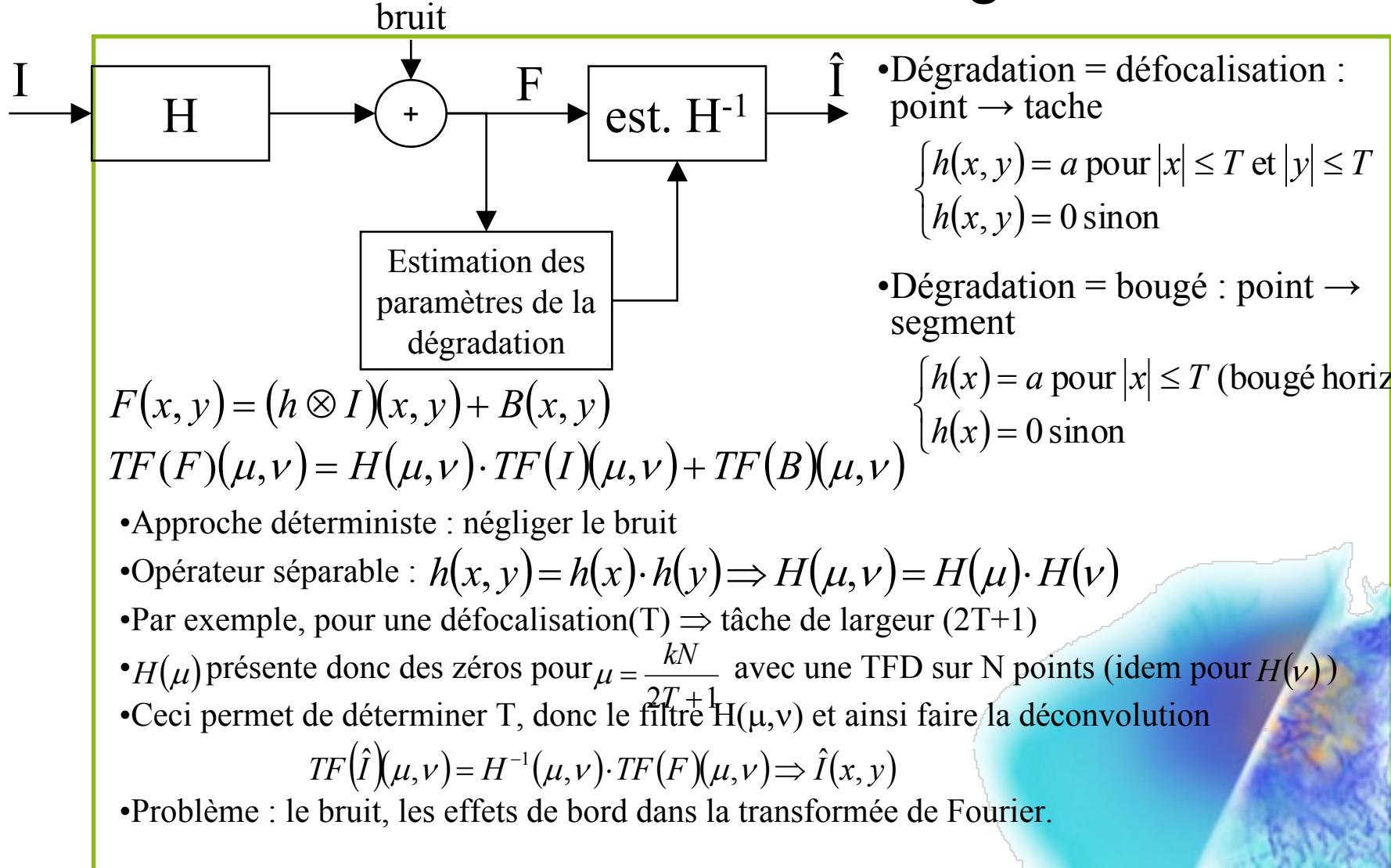
Filter Coefficients



Frequency Response



Restauration d'images



Traitement des images par outils de morphologie mathématique

- Images binaires, images de niveaux de gris
 - distances
 - érosion/dilatation
 - ouverture/fermeture
 - Filtrage
- Bases de la morpho - mathématique (J.P. Serra 1966)
 - min. : érosion(I)
 - max. : dilatation(I)
 - ouverture : dilatation(érosion(I))
 - fermeture : érosion(dilatation(I)))
 - dilatation - érosion : contour

Métriques

- Définitions :

$$p(x,y); q(s,t); r(u,v)$$

$$D(p,q) = 0 \text{ si } p = q$$

$$D(p,q) > 0 \text{ si } p \neq q$$

$$D(p,q) = D(q,p)$$

$$D(p,r) \leq D(p,q) + D(q,r)$$

Distances

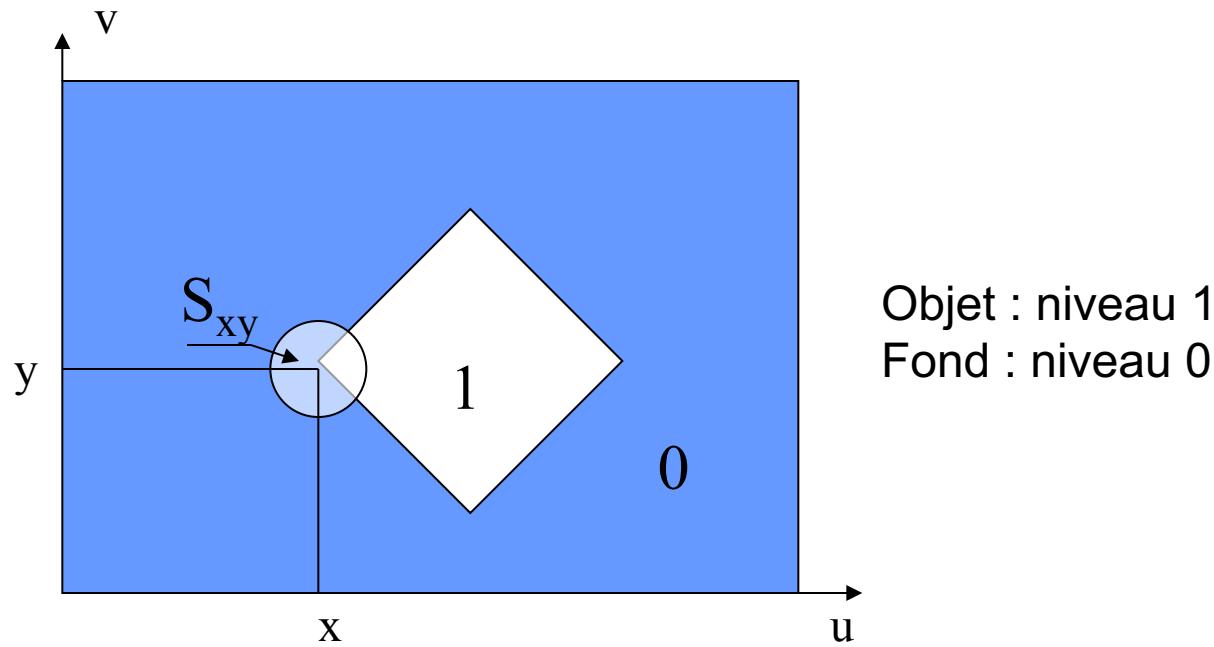
$$\text{Euclidienne (H}_2\text{): } D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$$

$$\text{4--voisins (H}_1\text{): } D_4(p,q) = |x-s| + |y-t|$$

$$\text{8--voisins (H}_{\infty}\text{): } D_8(p,q) = \max[|x-s|, |y-t|]$$

Morphomath : érosion & dilatation

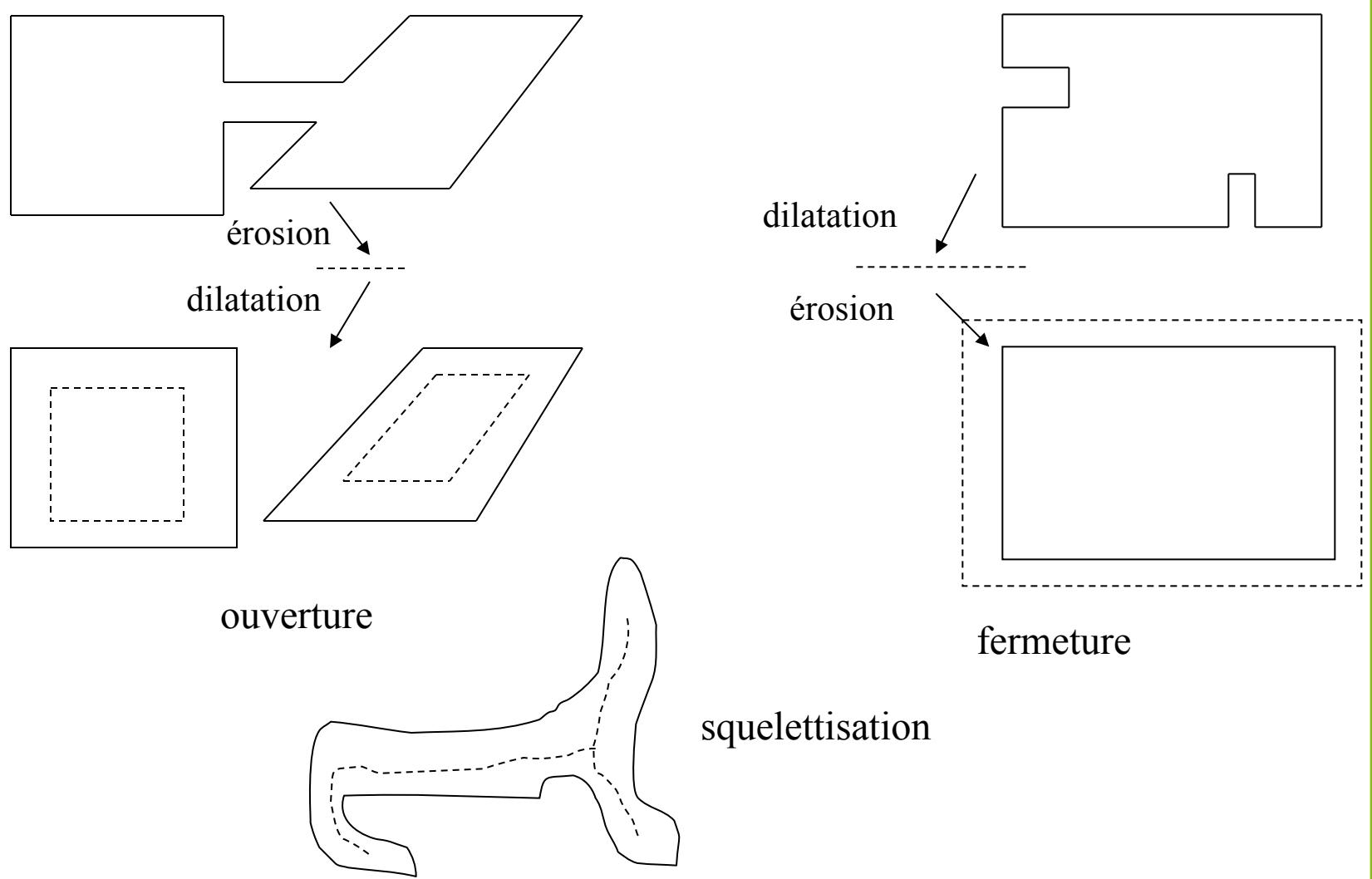
- S_{xy} : élément structurant placé en (x,y) , en g^{al} une « boule »
- $B(u,v)$: image initiale, $B'(u,v)$ image résultante
 - Erosion : $B'(x,y) = \min_{(a,b) \in S_{xy}} [B(u,v)]$
 - Dilatation : $B'(x,y) = \max_{(a,b) \in S_{xy}} [B(u,v)]$



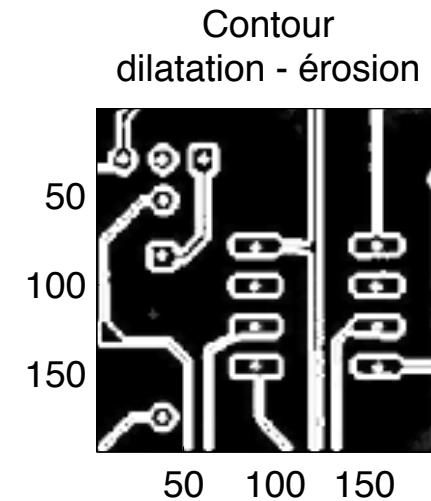
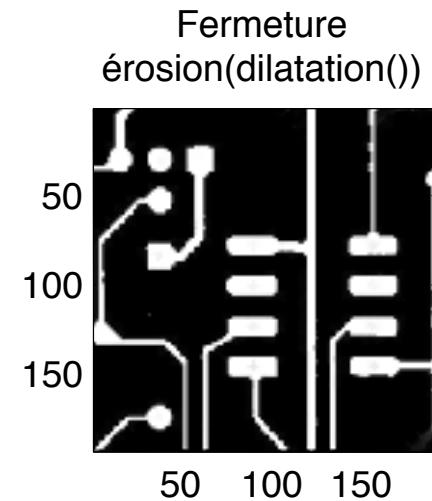
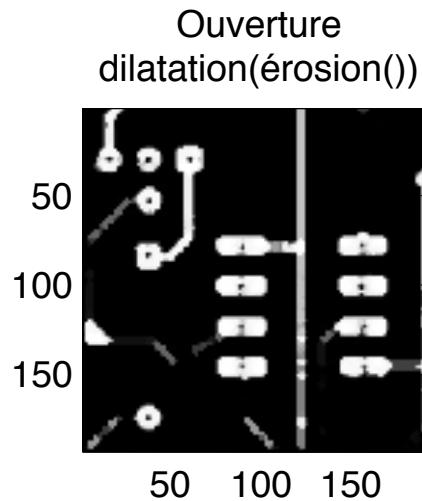
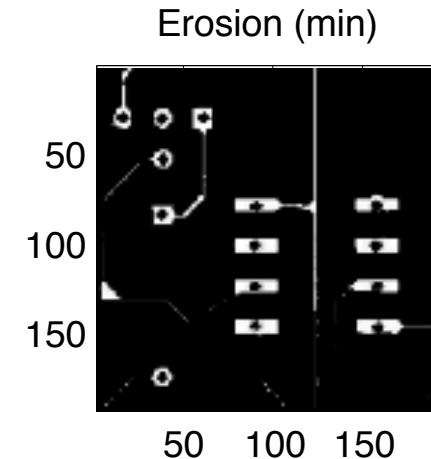
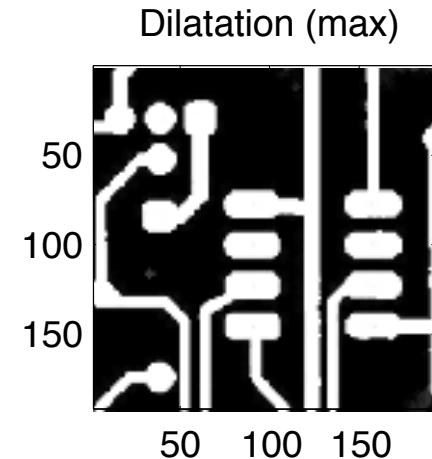
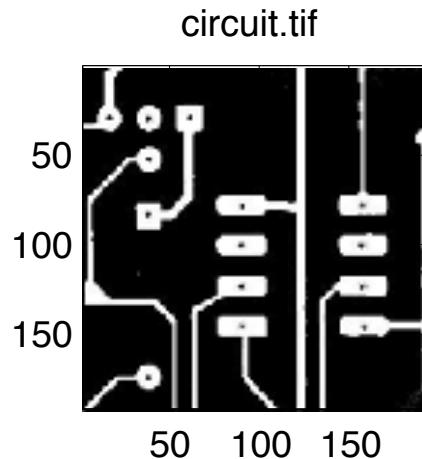
Morphologie mathématique

- Erosion, dilatation
- Ouverture
 - érosions, dilatations : dilatation(érosion(I))
- Fermeture
 - dilatations, érosions : érosion(dilatation(I)))
- squelettisation
 - érosion qui préserve la connexité
- filtrage sur image non binaires
 - base: érosion-dilatation ou ouverture-fermeture

Morphologie mathématique: exemples



Opérations de Morpho-Math



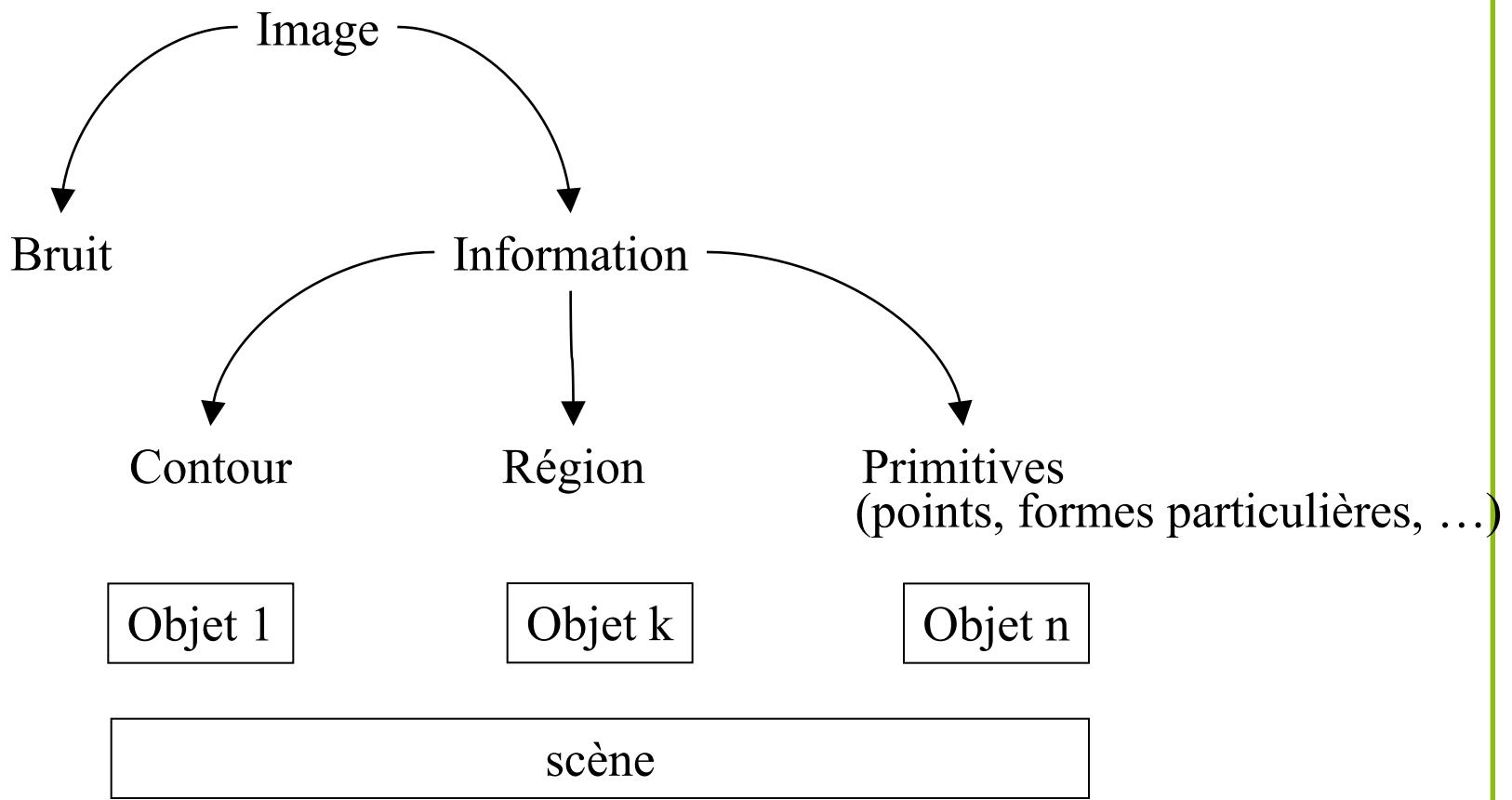
Analyse des images : Segmentation

Recherche de primitives d'intérêt

- régions
 - niveau de gris
 - couleur
 - texture
 - autres: zones de flot optique homogène (vecteur caractérisant le déplacement d'un objet), fréquences homogènes, etc.
- contours
 - détection des frontières entre régions
 - amincissement des contours, fermeture de contours
 - recherche de primitives géométriques sur les contours: segments, autres
- motifs particuliers: points, formes
 - détection et localisation

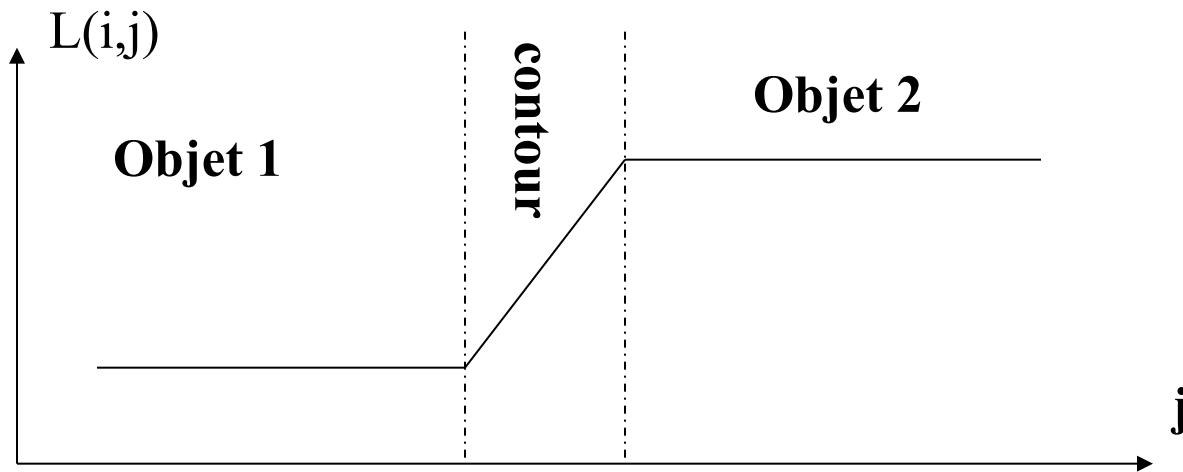
Analyse des images

- Caractérisation des images



Analyse des images

- Modèle de
 - distorsions spatiales (capteur)
 - flou (bougé, mise au point)
 - bruit
- Modèle de contour
 - séparation entre deux régions voisines de l'image caractérisées par une variation de niveau de gris, ou de couleur, texture.....

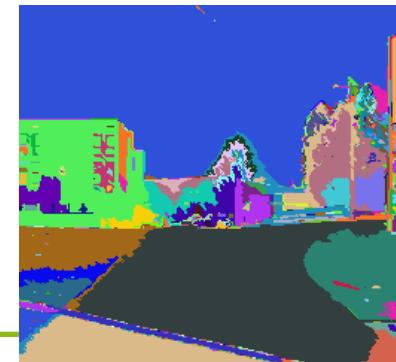
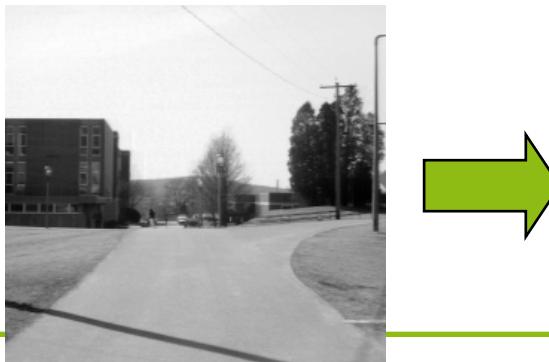


Segmentation : Différentes approches

- Détection de discontinuités
- seuillage
- approche région
- morphologie mathématique (ligne de partage des eaux - watershed)

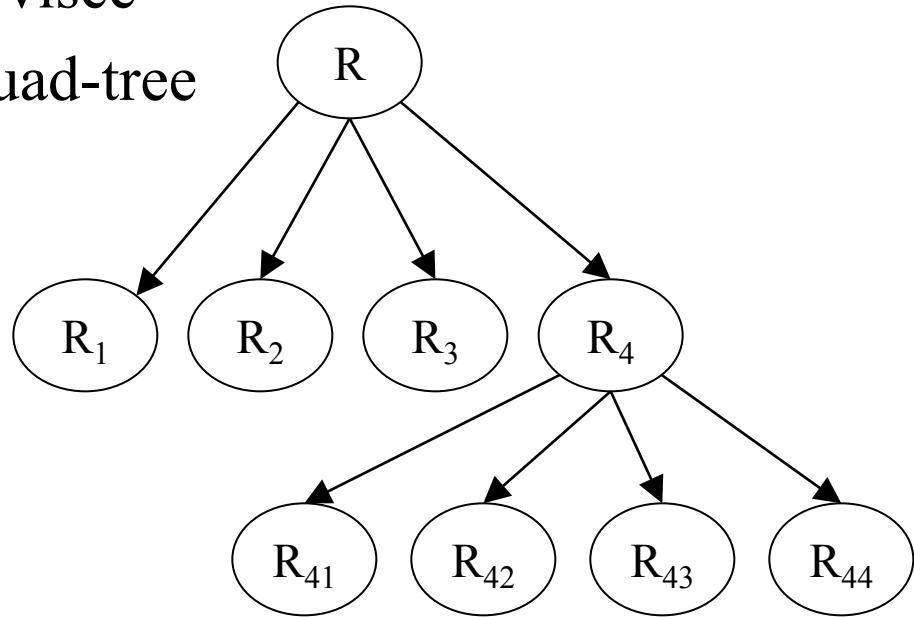
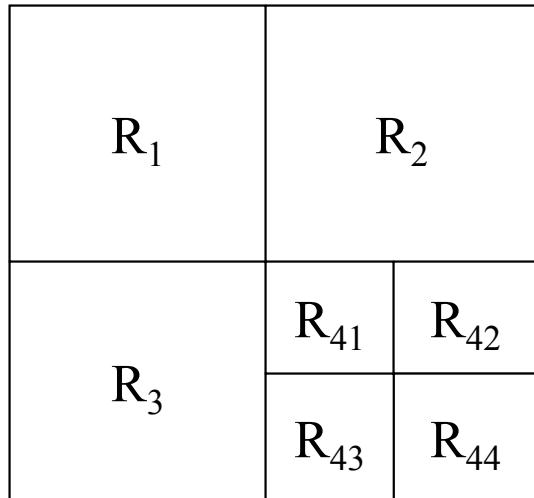
Segmentation : approche région

- Formulation de base
 - R : image entière. $R_i \ i \in [1 \dots n]$ sous ensembles de R
 - $P(R_i)$ prédicat logique défini sur tout R_i
 - ex1 : les niveaux de gris de R_i sont compris entre S_{1i} et S_{2i}
 - Si $P(R_i)$ est vrai alors les niveaux de gris de R_i sont fixés à z_i / $S_{1i} \leq z_i \leq S_{2i}$
 - ex2 : 80% des niveaux de gris de R_i sont / $|z_j - m_i| \leq 2\sigma_i$
 - z_j niveau de gris du $j^{\text{ème}}$ pixel de R_i , (m_i, σ_i) niveau de gris moyen et écart type dans R_i
 - Si $P(R_i)$ vrai alors les niveaux de R_i sont fixés à $z_i = m_i$
- Segmentation = partition de R en $R_i \in [1 \dots n]$ de telle manière que
 - $\bigcup_{i=1}^n R_i = R$
 - R_i est un ensemble de pixels connexes satisfaisant un prédicat $P(R_i)$
 - $R_i \bigcap R_j = \text{ensemble vide}$
 - $P(R_i) = \text{vrai}$
 - $P\left(R_i \bigcap \bigcup_{i \neq j} R_j\right) = \text{faux}$



Segmentation: approche région

- Croissance de région
- Classification non supervisée
- Séparation-fusion par quad-tree



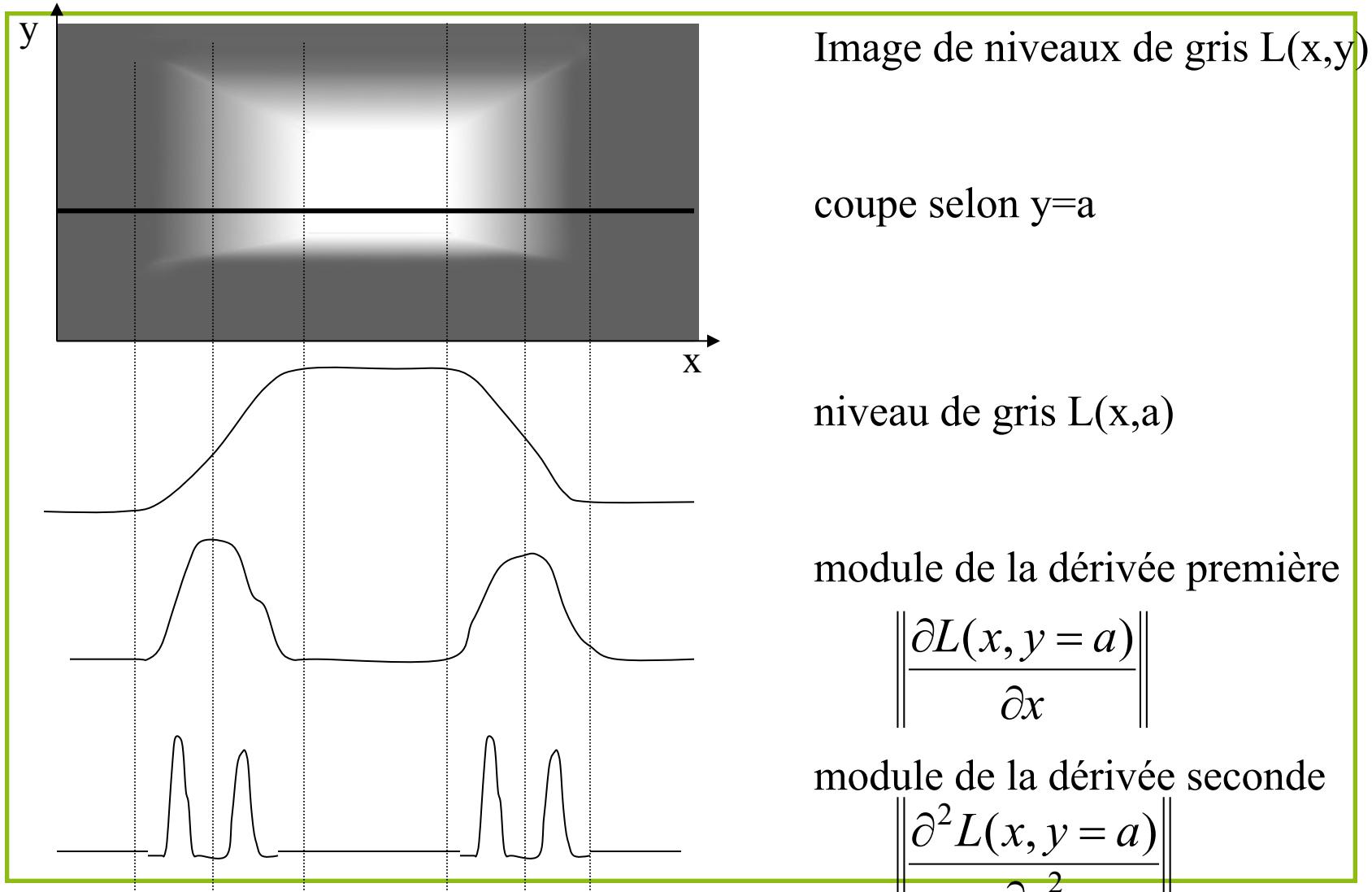
partitionner R en quadrants disjoints / $P(R_i)$ vrai

fusionner R_i et R_k adjacents si $P(R_i \cup R_k)$ vrai

arrêter selon critère de fin (taille minimale des régions par exemple)

Segmentation: détection de discontinuités

Opérateurs gradients et laplacien



Opérateurs gradients et laplaciens

Image de luminance $L(x, y)$

Vecteur gradient : $G(x, y) = (G_x, G_y)^T$

$$G_x = \frac{\partial L(x, y)}{\partial x} , \quad G_y = \frac{\partial L(x, y)}{\partial y}$$

normes du gradient :

$$G_1 = \|G_x\| + \|G_y\|$$

$$G_2 = (G_x^2 + G_y^2)^{1/2}$$

$$G_\infty = \max[\|G_x\|, \|G_y\|]$$

propriété : le vecteur gradient est normal aux courbes iso - luminances

Laplacien :

$$La(x, y) = \frac{\partial^2 L(x, y)}{\partial x^2} + \frac{\partial^2 L(x, y)}{\partial y^2}$$

Le laplacien est un scalaire

Opérateurs gradients et laplaciens à RIF

Gradient en croix : $h_x = \begin{vmatrix} 0 & 0 & 0 \\ -1 & 0 & +1 \\ 0 & 0 & 0 \end{vmatrix}$ $h_y = \begin{vmatrix} 0 & +1 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{vmatrix}$

Opérateurs de Robert $h_1 = \begin{vmatrix} 0 & -1 \\ 1 & 0 \end{vmatrix}$ $h_2 = \begin{vmatrix} -1 & 0 \\ 0 & 1 \end{vmatrix}$

Opérateurs de Prewitt ($c=1$) et de Sobel ($c=2$) : $h_x = \begin{vmatrix} -1 & 0 & +1 \\ -c & 0 & +c \\ -1 & 0 & +1 \end{vmatrix}$ $h_y = \begin{vmatrix} 1 & c & 1 \\ 0 & 0 & 0 \\ -1 & -c & -1 \end{vmatrix}$

Opérateurs de Kirsh :

$$h_1 = \begin{vmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{vmatrix} \quad h_2 = \begin{vmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{vmatrix} \quad h_3 = \begin{vmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{vmatrix} \dots h_8 = \begin{vmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{vmatrix}$$

8 opérateurs selon les 8 directions possibles → Amplitude du gradient = $\max_{i \text{ de } 1 \text{ à } 8} [I \otimes h_i]$

Opérateurs laplaciens

$$L_1 = \begin{vmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{vmatrix} \quad L_2 = \begin{vmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{vmatrix} \quad L_3 = \begin{vmatrix} +1 & -2 & +1 \\ -2 & 4 & -2 \\ +1 & -2 & +1 \end{vmatrix}$$

Recherche de contour après un gradient

- Gradient seul
- Gradient + laplacien
 - gradient: détection par seuil
 - laplacien: localisation par passage à 0
- Amincissement (et localisation fine) des contours:
 - recherche du maximum du gradient
 - opérateurs de morphologie mathématique
 - ligne de partage des eaux
- Fermeture:
 - seuillage par hystérésis
 - algorithmes de fermeture (théorie des graphes,)

Gradients et laplaciens (suite)

- Les opérateurs gradients et laplaciens sont des filtres linéaires non récursifs. Ils peuvent être associés à des filtres lisseurs.

I : image, g filtre lisseur, \otimes convolution

$I \otimes g \rightarrow$ image lissée

gradient de l'image lissée $(I \otimes g)' = I \otimes g'$

laplacien de l'image lissée: $(I \otimes g)'' = I \otimes g''$

filtres séparables: $g(x,y) = g(x).g(y) \Rightarrow (I \otimes g_x) \otimes g_y$

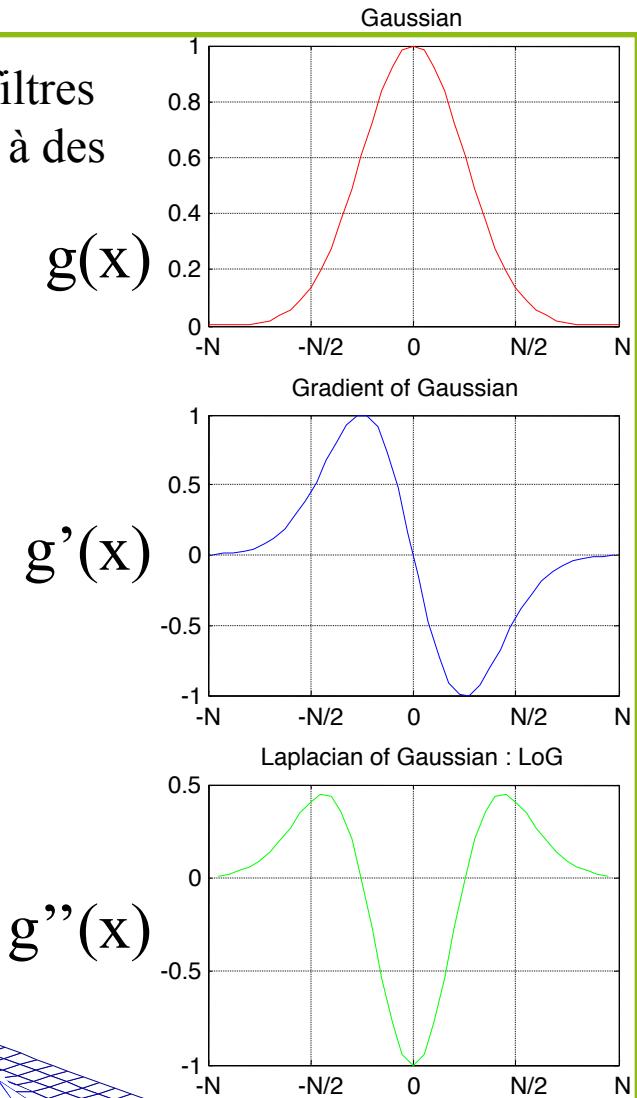
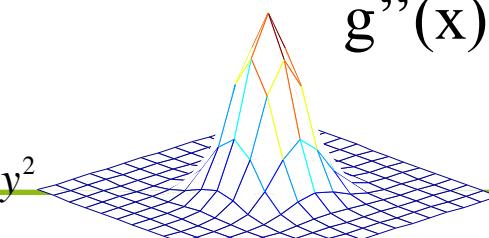
ex: filtre de lissage gaussien:

$$g(x) = -ce^{-\frac{x^2}{2\sigma^2}}, g(y) = -ce^{-\frac{y^2}{2\sigma^2}}$$

$$g'_x(x) = +c \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}}$$

$$g''_x(x) = -c \left(\frac{x^2 - \sigma^2}{\sigma^4} \right) e^{-\frac{x^2}{2\sigma^2}}$$

en deux dimensions : $x^2 \rightarrow r^2 = x^2 + y^2$



Opérateurs déivateurs

- Gradient : $I_x(x, y) = \frac{\partial I(x, y)}{\partial x}$, $I_y(x, y) = \frac{\partial I(x, y)}{\partial y}$

$$G(x, y) = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) = (I_x, I_y), |G| = \sqrt{I_x^2 + I_y^2}, \Phi = \text{atan} \left(\frac{I_y}{I_x} \right)$$

- Laplacien : $L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$

- Filtres classiques

– Prewitt

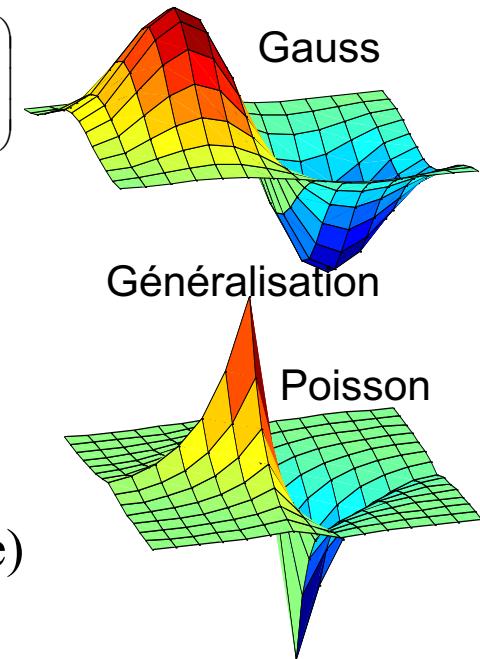
$$\begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad L$$

– Sobel

$$\begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \quad I_x \quad I_y$$

- Filtrages évolués

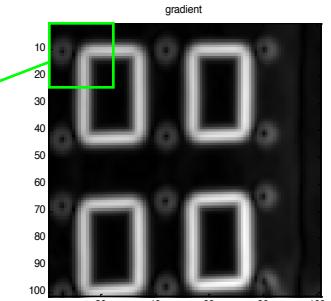
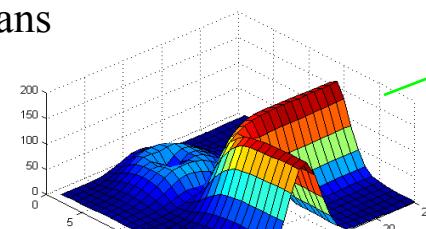
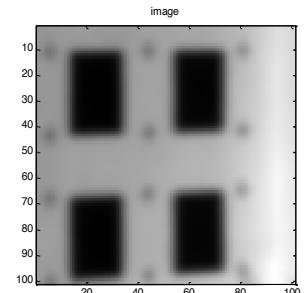
- non linéarité
- récursivité
- séparabilité
- exemple : Canny-Deriche (récursif / séparable)



Détection de contours

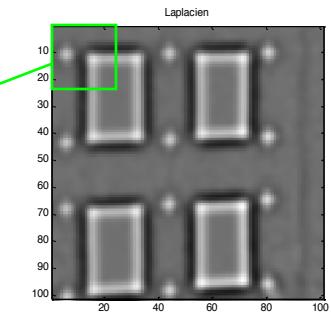
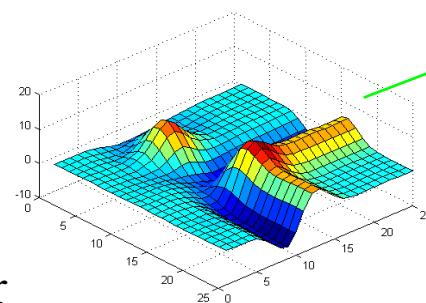
- Approche Gradient : recherche de maxima

- Calcul de la norme du gradient
 - Sélection des points de plus fort gradient
 - Extraction des extrema locaux dans la direction du gradient
 - Recherche d'extrema locaux connexes dans la direction \perp au gradient (pour la fermeture)



- Approche Laplacien : recherche de passages par zéro

- Calcul du Laplacien
 - Sélection des points de chgt de signe
 - Recherche des + forts passages par zéro connexes



Détection de points d'intérêt

- Exemple : Mesure de Cornerness de Beaudet

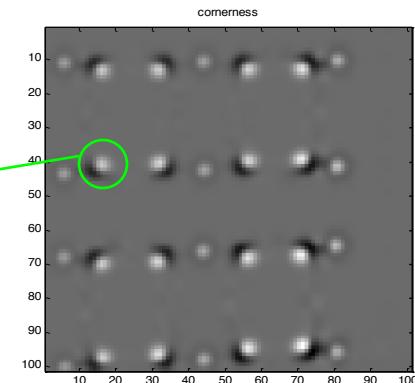
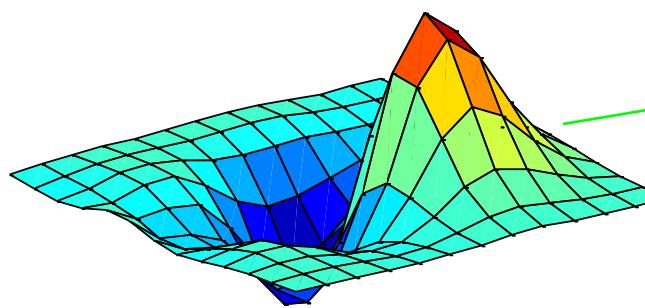
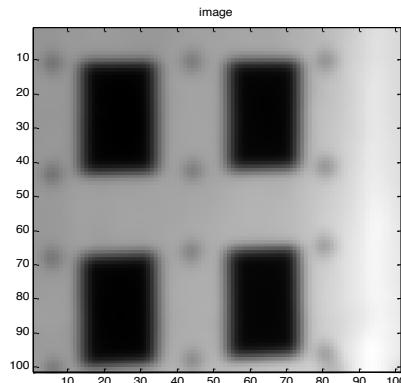
- Combinaisons linéaires de dérivées de l'image

- Mesure de Beaudet : $c(I) = \left(\frac{\partial^2 I}{\partial x^2} \cdot \frac{\partial^2 I}{\partial y^2} \right) - \left(\frac{\partial I}{\partial x \partial y} \cdot \frac{\partial I}{\partial y \partial x} \right)$

- Autre mesure :

$$c(I) = \frac{\left(\frac{\partial^2 I}{\partial x^2} \cdot \left(\frac{\partial I}{\partial y} \right)^2 + \frac{\partial^2 I}{\partial y^2} \cdot \left(\frac{\partial I}{\partial x} \right)^2 - \left(\frac{\partial I}{\partial x \partial y} \cdot \frac{\partial I}{\partial x} \cdot \frac{\partial I}{\partial y} \right)^2 \right)}{\left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2}$$

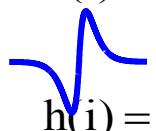
- Localisation optimale entre le pic et le creux



Filtre déivateur récursif de Canny-Deriche

- Fonctions et propriétés recherchées (critères de Canny)
 - détection: l'opérateur doit donner une réponse au voisinage d'un contour
 - localisation: le contour doit être localisé avec précision
 - réponse unique: le contour doit provoquer une seule réponse de l'opérateur d'extraction
 - optimisation: en fonction du type de discontinuité et du type de bruit

$$h(i) = cie^{-\alpha|i|}$$



$$H(z) = \sum_{i=-\infty}^{+\infty} h(i)z^{-i}$$

$$h(i) = h_-(i) + h_+(i)$$

$$h_-(i) = \begin{cases} cie^{-\alpha i} & i \geq 0 \\ 0 & i < 0 \end{cases} \quad h_+(i) = \begin{cases} 0 & i \geq 0 \\ cie^{+\alpha i} & i < 0 \end{cases}$$

$$H(z) = H_-(z^{-1}) + H_+(z^{-1})$$

$$H_-(z^{-1}) = \frac{ce^{-\alpha} z^{-1}}{1 - 2e^{-\alpha} z^{-1} + e^{-2\alpha} z^{-2}} = \frac{Y_1}{X} \quad H_+(z^{-1}) = \frac{-ce^{-\alpha} z}{1 - 2e^{-\alpha} z + e^{-2\alpha} z^2} = \frac{Y_2}{X}$$

$$y_1(i) = ce^{-\alpha} x(i-1) + 2e^{-\alpha} y_1(i-1) - e^{-2\alpha} y_1(i-2) \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{Equations récurrentes}$$

$$y_2(i) = -ce^{-\alpha} x(i+1) + 2e^{-\alpha} y_2(i+1) - e^{-2\alpha} y_2(i+2) \quad \left. \begin{array}{l} \\ \end{array} \right\}$$

$$y(i) = y_1(i) + y_2(i)$$

Deriche, R.. - *Using Canny's criteria to derive a recursively implemented optimal edge detector.*
International Journal of Computer Vision . Springer, , 2 , n° 2, 167-187 (June 1987)

Dualité contours/régions

- Toute région doit être bordée par un contour fermé
 - Néanmoins les méthodes de détection des deux primitives sont radicalement différentes !
- Contours : détection locale
- Régions : construction « globale »
- La dualité n'est assurée que lorsque les deux segmentations coopèrent
 - Exemples :
 - contours comme critères d 'arrêt de croissance des régions
 - utilisation bords de régions pour fermer les contours



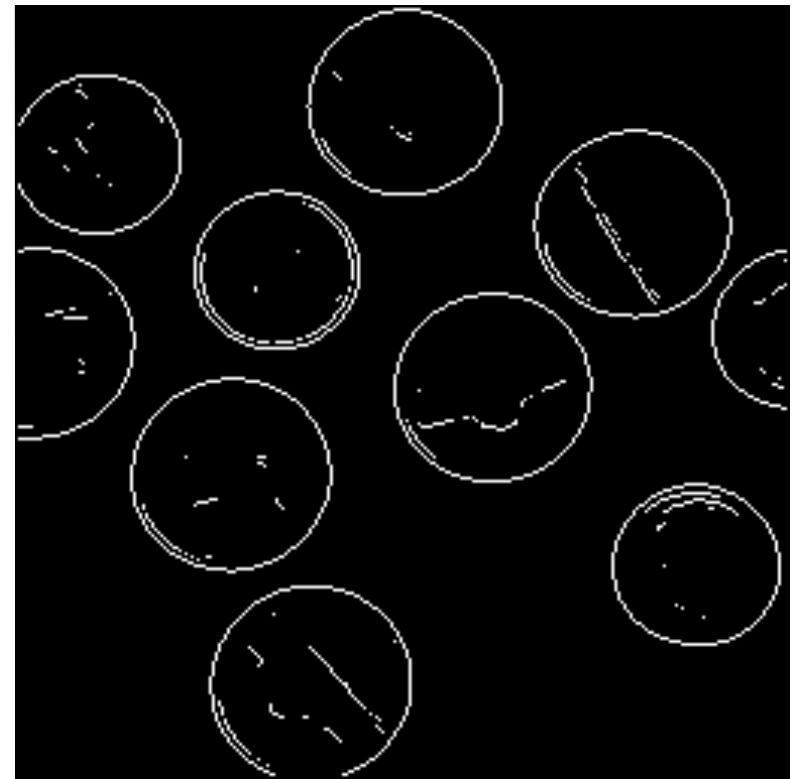
Détection de contours

- Filtre Sobel (X & Y), seuillage de gradient

Original Coins Image



Edge Map



Détection de contours

- Filtre LoG, seuillage de la pente des passages par zéro

Original Coins Image



Edge Map



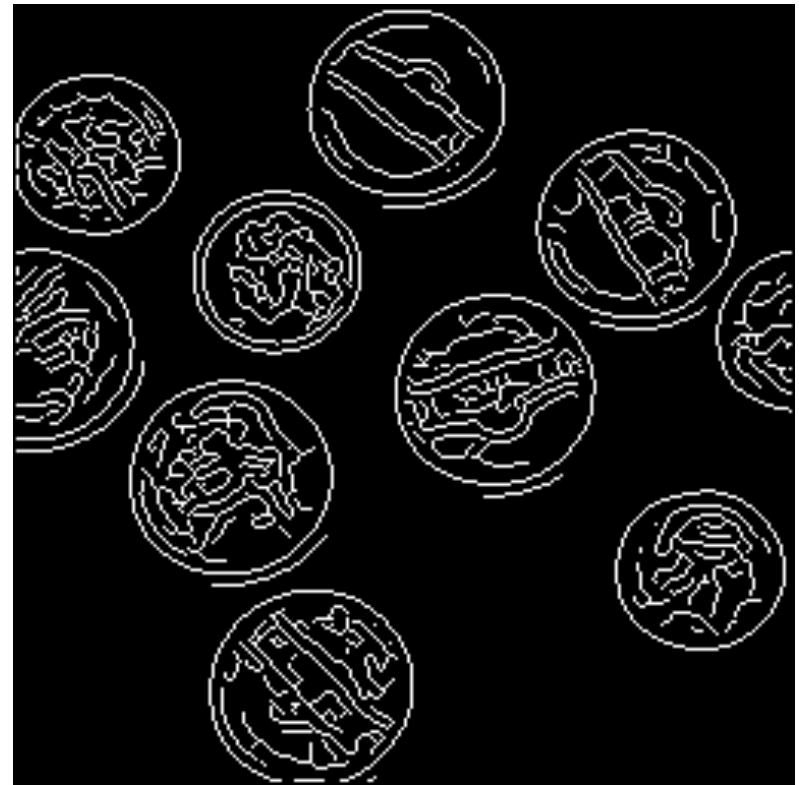
Détection de contours

- Filtre Canny (DoG), seuillage du gradient

Original Coins Image



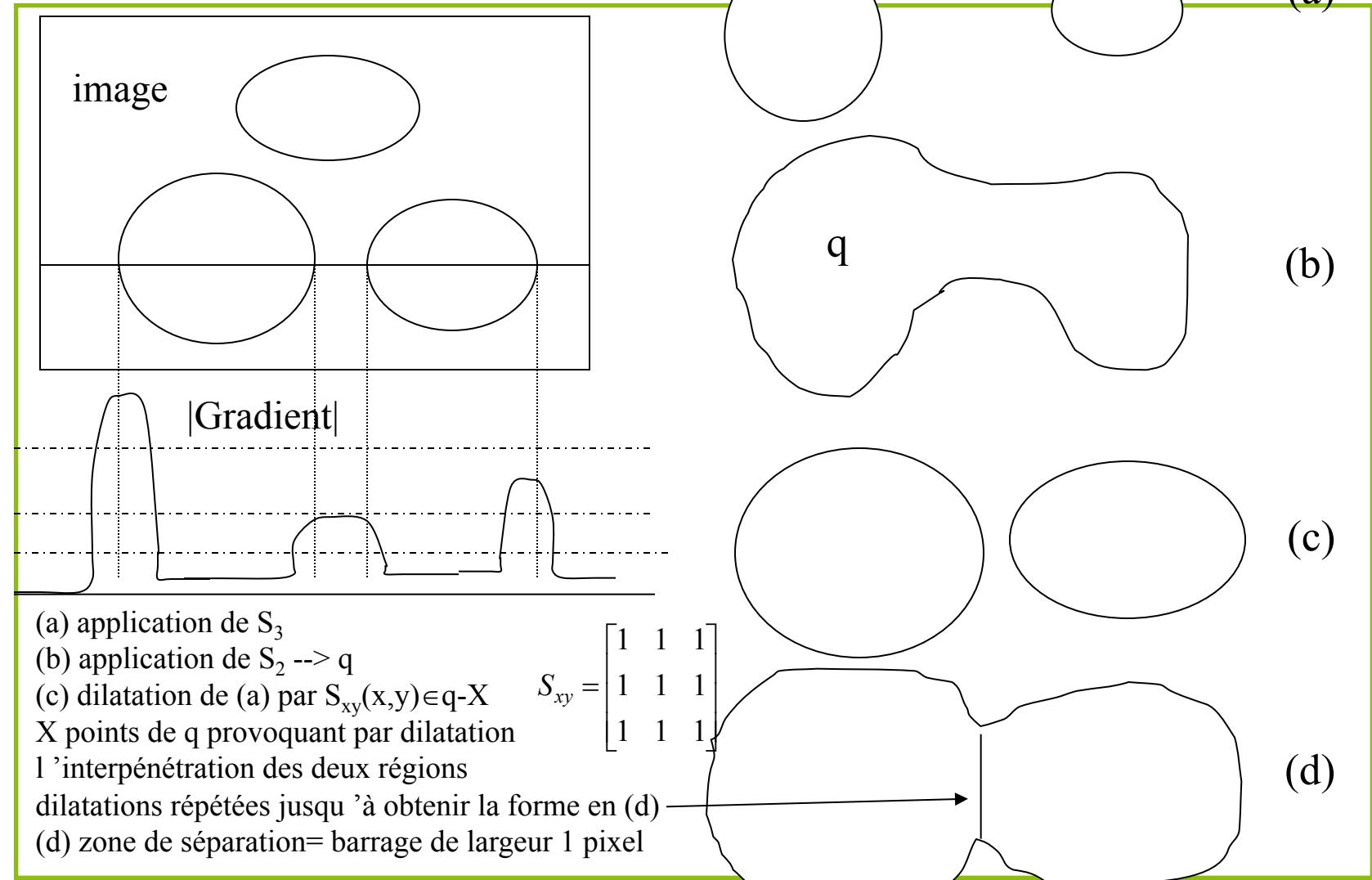
Edge Map



Segmentation: partage des eaux par morphologie mathématique

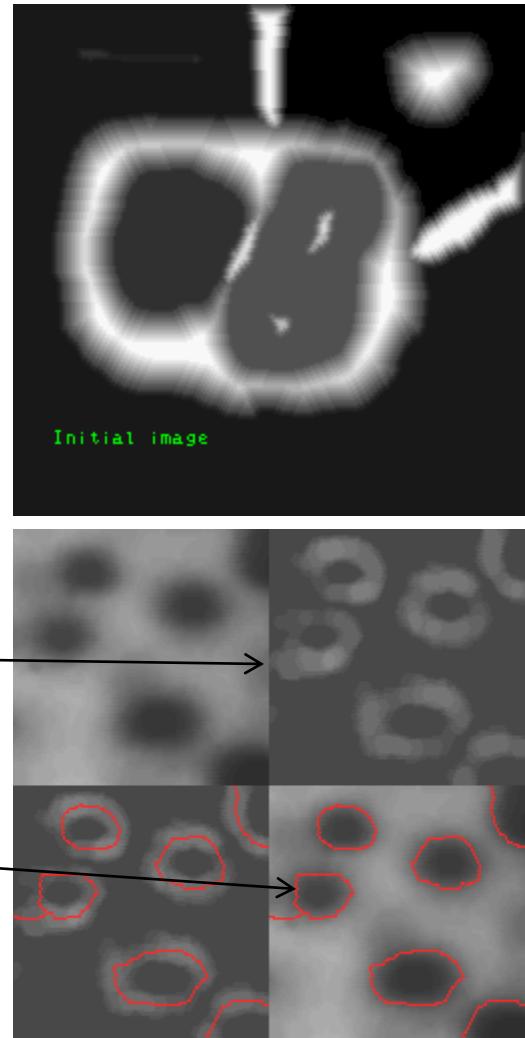
- objectif: séparer les régions par un contour continu et fermé, d'épaisseur 1 pixel
- image de départ: image de gradient
- Gonzalez et Woods. Digital Image Processing. pp 617-.....

ligne de partage des eaux: principe



Lignes de partage des eaux (Watershed lines)

- Principe :
 - Considérer une image comme une surface topographique
 - En inondant cette surface à partir de ses minima tout en évitant le mélange des eaux de différentes sources, on partitionne l'image en deux ensembles :
 - Les bassins versants
 - Les lignes de partage des eaux
- Application :
 - En appliquant ce traitement aux images de gradient
 - Les bassins versants correspondent aux régions homogènes de l'image
 - Les lignes de partage des eaux correspondent aux contours de ces régions



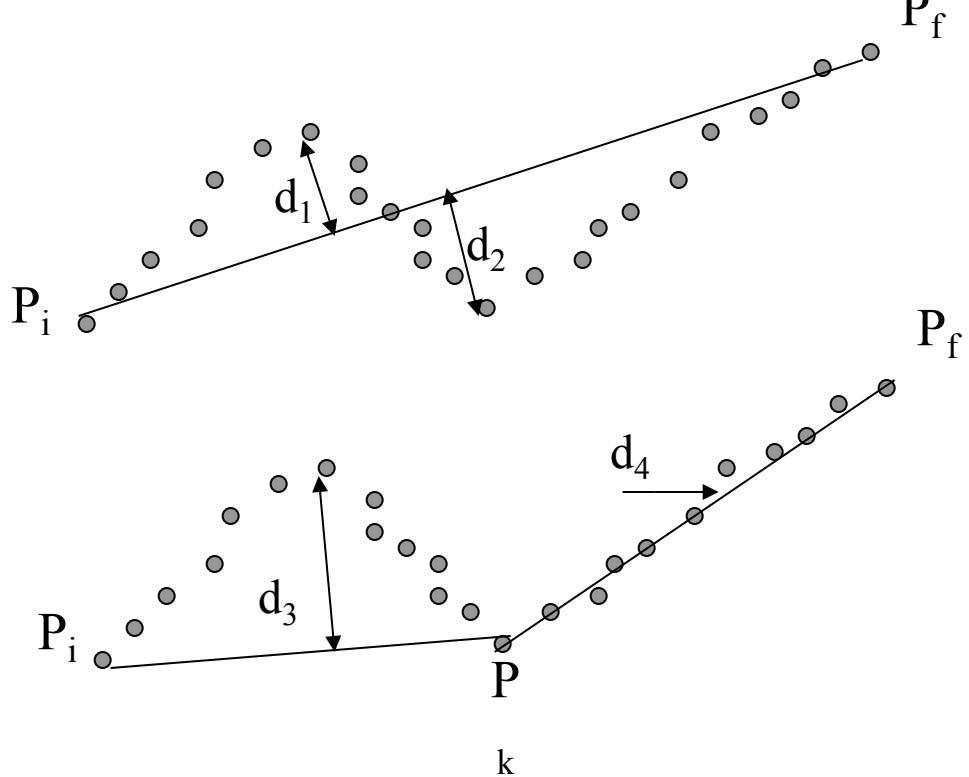
Source : Serge Beucher - Centre de Morphologie Mathématique
Mines - ParisTech

Contours polygonaux

- chaîne de points \in contour \rightarrow segments
 - algorithme de Duda-Hart
 - R.O. Duda, P.E. Hart. « Pattern classification and scene analysis », New-York, Wiley, 1973
 - algorithmes de traitements séquentiels
 - Douglas, D. H. and Peucker, T. K. « Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or Its Caricature », *Cartographica: The International Journal for Geographic Information and Geovisualization* 10 112--122 (1973)
 - K. Wall, P.E. Danielson. « A fast sequential method for polygonal approximation digitized curves », *CVGIP*, 28, 1984, pp 220-227.
 - transformée de Hough
 - R.O. Duda, P.E. Hart. « Use of the Hough transform to detect lines and curves in pictures », *Com. ACM*, Vol 15, 1972, pp 11-15.
- affinement par les moindres carrés
- regroupement/élimination de segments

Algorithme de Duda-Hart*

- algorithme récursif



$P_{\text{initial}} = P_i$, $P_{\text{final}} = P_f$

d_1 et $d_2 >$ seuil

$d_1 < d_2 \rightarrow P_{\text{initial}} = P_k$



$P_{\text{initial}} = P_k$, $P_{\text{final}} = P_f$

$d_4 <$ seuil $\rightarrow P_k P_f = \text{segment}$



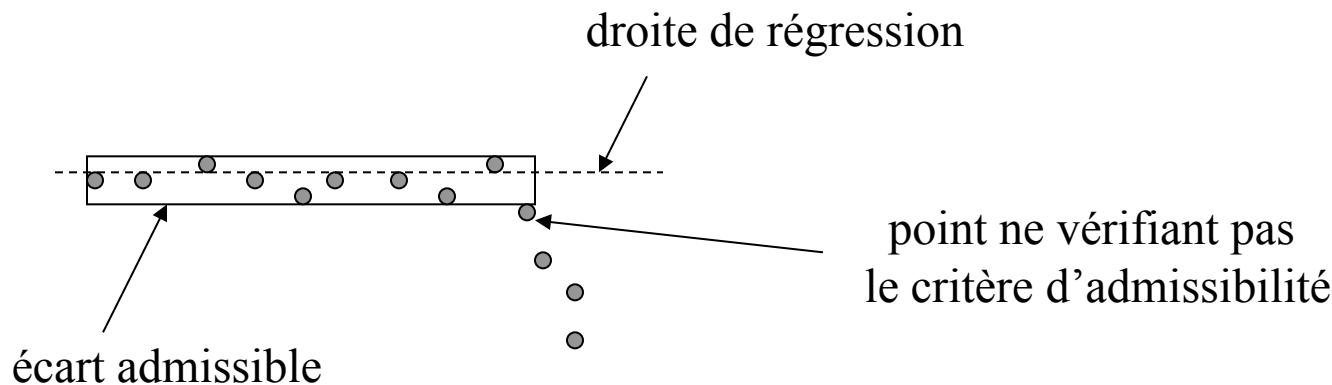
$P_{\text{initial}} = P_i$, $P_{\text{final}} = P_k$

.....

* En fait plutôt de Douglas & Peucker

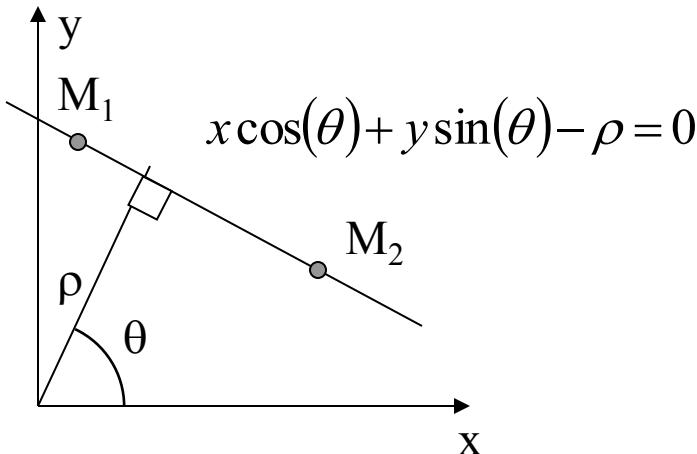
Algorithme de traitement séquentiel des points

- un point nouveau sur la chaîne est intégré au segment courant s'il vérifie un critère d'admissibilité, sinon on créé un nouveau segment dont le point initial est celui-ci .
 - algorithmes d'identification de paramètres de type moindres carrés récursifs
 - autres,... exemple: algorithme de Wall-Danielson

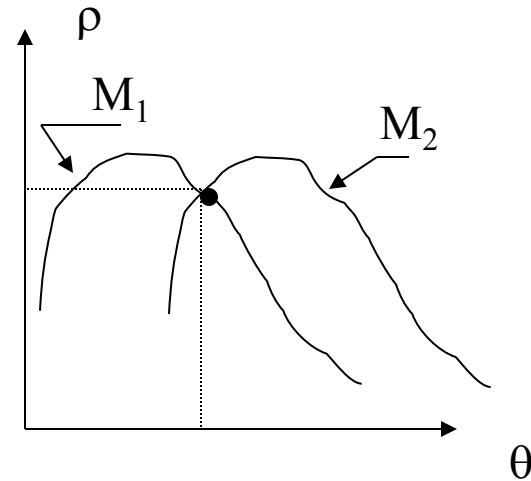


Transformée de Hough

- transposition coordonnées cartésiennes → coordonnées paramétriques
- algorithme d'accumulation de points dans l'espace des paramètres



coordonnées cartésiennes



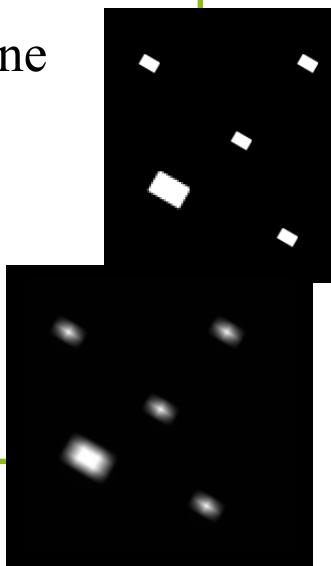
coordonnées paramétriques

Localisation de primitives géométriques

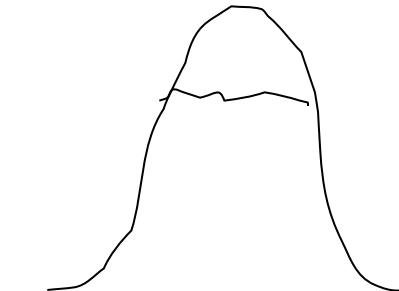
- Objectif: localisation sub-pixellique afin de faire des mesures 2D les plus précises possibles
 - applications: métrologie 2D, calibration de caméras, reconstruction 3D, etc.
- primitives concernées: point , tache, croix,

Recherche d'une primitive par corrélation

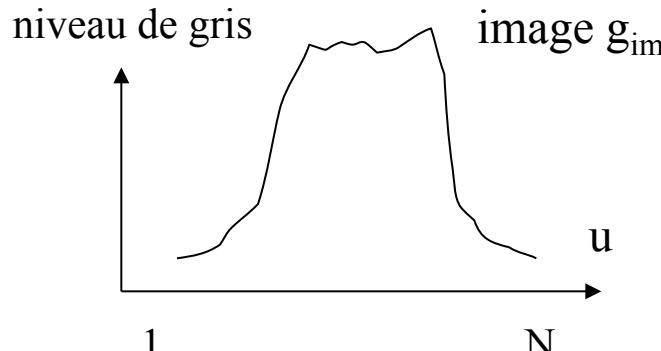
- Corrélation bi dimensionnelle
 - mesure de similitude entre deux signaux : $x(k,l)$ et $y(k,l)$ pour des signaux dont l'un au moins est à étendue infinie
 - maximum de la mesure lorsque la ressemblance entre $x(k,l)$ et $y(k,l)$ est la plus grande (masques)
 - Utilisation en analyse d'image : détection d'un objet dans une image :
 - $x(k,l)$: image
 - $y(k,l)$: objet recherché
 - Applications: reconnaissance des caractères, recherche de primitives, suivi d'objet...



Localisation d'un point



maximum mal défini
barycentre de la tache $\rightarrow 0,1$ pixel
méthode de corrélation $\rightarrow 0,1$ à $0,01$ pixel



$$\text{Min}_{a,b,c,\sigma} \sum_{u=1}^{u=N} [(bg_r)^\sigma(u - a) + c] - g_{im}(u)]^2 \rightarrow a \text{ position du point}$$

Courbes références candidates: gaussienne, portion de cosinus,

En deux dimensions, la corrélation est faite sur (u,v)

Pour réduire l'espace de recherche faire une première estimation
de la position du point et de la fenêtre de recherche

La texture

- Attribut d'une région de l'image qui n'est ni de niveau de gris uniforme, ni de couleur uniforme.
- Différents types de modélisation adaptées aux différents types de textures
 - techniques stochastiques
 - techniques structurelles
 - répétition de motifs de base suivant certaines directions
 - représentation hiérarchique (motif+règle de placement)
 - techniques fréquentielles
 - zones de fréquences homogènes
 - périodicités

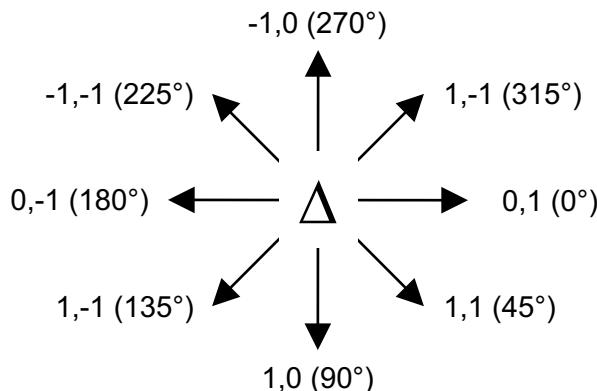
La texture: approche statistique

- Moments : $\mu_n = \sum_{i \in R} (z_i - m)^n p(z_i)$
- valeur moyenne : $m = \sum_{i \in R} z_i p(z_i)$
- z niveaux de gris
- moments utilisés : $\mu_2 = \sigma^2$, μ_3
- autres mesures : entropie moyenne : $E = -\sum_{i \in R} p(z_i) \log_2(p(z_i))$
- Différentes descriptions
 - P. ex. : Matrice de co-occurrences : Soit $\Delta_{k,\theta}$ un vecteur de translation dans l'image (k amplitude (en g^{al} k=1)), θ angle)
 - La matrice de co-occurrence traduit la probabilité qu'un pixel ait le niveau i et que son translaté ait le niveau j
 - Matrice carrée de taille N×N (N = nombre de niveaux de gris)
 - Autant de matrices qu'il y a de vecteurs Δ

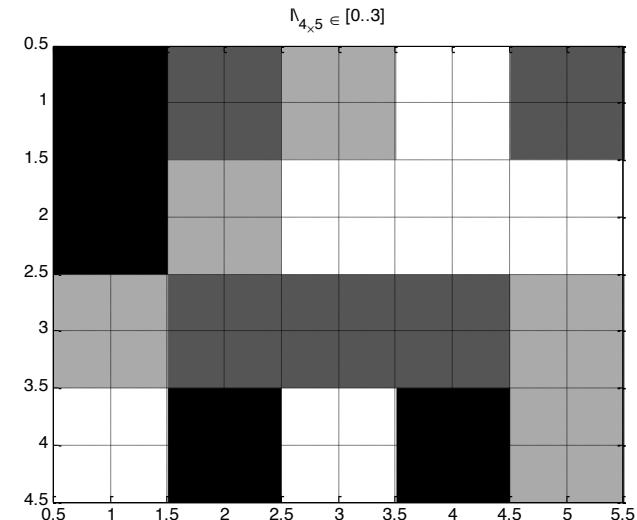
Haralick, R. M., Shanmugam, K. & Dinstein, I.. - *Textural features for image classification* . IEEE Transactions on Systems, Man, and Cybernetics , SMC-3 , n° 6, 610-621 (November 1973)

Matrices de cooccurrences (run-length)

- Image : $I_{4 \times 5}(i,j)$ avec $(i,j) \in [0..3]$



0
1
2
3



- Matrices de cooccurrences ($\text{nb}_{\text{niveaux}} \times \max(\text{size}(I))$)

Level	0°	1	2	3	4	5
0	4	0	0	0	0	0
1	2	0	1	0	0	0
2	5	0	0	0	0	0
3	3	0	1	0	0	0

45°	1	2	3	4	5
0	4	0	0	0	0
1	5	0	0	0	0
2	5	0	0	0	0
3	4	1	0	0	0

90°	1	2	3	4	5
0	2	1	0	0	0
1	5	0	0	0	0
2	3	1	0	0	0
3	4	1	0	0	0

Galloway, Mary M.. - *Texture analysis using gray level run lengths*. Computer Graphics and Image Processing , Vol 4 , n° 2, 172-179 (June 1975)

Mesures sur les matrices de co-occurrence

- Notations
 - $p(i,j)$: (i,j) ème élément de la matrice de run-length
 - N_g le nombre de niveaux de gris
 - N_r les différentes longueurs de run
 - P le nombre de pixels de l'image
- Short Runs emphasis :
$$\sum_{i=1}^{N_g} \sum_{j=1}^{N_r} \frac{p(i,j)}{j^2} \Bigg/ \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} p(i,j)$$
- Long Runs emphasis :
$$\sum_{i=1}^{N_g} \sum_{j=1}^{N_r} j^2 p(i,j) \Bigg/ \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} p(i,j)$$
- Gray Level Nonuniformity :
$$\sum_{i=1}^{N_g} \left(\sum_{j=1}^{N_r} p(i,j) \right)^2 \Bigg/ \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} p(i,j)$$
- Run Length Nonuniformity :
$$\sum_{j=1}^{N_r} \left(\sum_{i=1}^{N_g} p(i,j) \right)^2 \Bigg/ \sum_{i=1}^{N_g} \sum_{j=1}^{N_r} p(i,j)$$
- Run Percentage :
$$\sum_{i=1}^{N_g} \sum_{j=1}^{N_r} p(i,j) \Bigg/ P$$

Vision dynamique : le flot optique (optical flow)

- Définition

- Pour chaque pixel d'une série d'images, on cherche le vecteur de vitesse de chaque point $\dot{U} = (\dot{u}, \dot{v})$

- Soit $I(u, v, t)$ la luminance de l'image dans le temps :

$$I(u + du, v + dv, t + dt) = I(u, v, t) + \frac{\partial I}{\partial u} du + \frac{\partial I}{\partial v} dv + \frac{\partial I}{\partial t} dt + \dots$$

- Soit une portion d'objet à la position (u, v) au temps t qui a bougé de (du, dv) après dt . Si on suppose que cette portion d'objet conserve sa luminosité initiale alors :

$$\begin{aligned} I(u + du, v + dv, t + dt) &= I(u, v, t) \\ \frac{\partial I}{\partial u} du + \frac{\partial I}{\partial v} dv + \frac{\partial I}{\partial t} dt + \dots &= 0 \end{aligned}$$

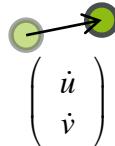
avec $\frac{du}{dt} = \dot{u}$ & $\frac{dv}{dt} = \dot{v}$

- Si $dt \rightarrow 0$, alors on a :
$$-\frac{\partial I}{\partial t} = \frac{\partial I}{\partial u} \dot{u} + \frac{\partial I}{\partial v} \dot{v}$$
 équation de contrainte du flot optique
- $\frac{\partial I}{\partial t}$, $\frac{\partial I}{\partial u}$ et $\frac{\partial I}{\partial v}$ peuvent être estimés d'après les images de la séquence

$I(u(t), v(t), t)$



$I(u(t+dt), v(t+dt), t+dt)$



Méthodes de détermination du flot optique

- Différentielles locales
 - Flot optique \leftarrow dérivées spatio-temporelles de l'image d'intensité.
 - Hypothèse de conservation de l'intensité dans le temps : variations d'intensité uniquement dues aux déplacements de l'image de l'objet observé et non à des problèmes de modifications de l'éclairage ou des réflexions sur l'objet en mouvement.
- Techniques d'appariements
 - Mise en œuvre quand les dt sont trop importants.
 - Deux images successives sont appariées, on en déduit immédiatement le vecteur flot optique pour les points appariés comme étant le déplacement de la paire appariée d'une image à l'autre.
- Approches fréquentielles
 - Deux signaux décalés dans le temps \rightarrow transformées de Fourier avec déphasage.

Détermination du flot optique par différentiation

$$\frac{\partial I}{\partial u} \dot{u} + \frac{\partial I}{\partial v} \dot{v} + \frac{\partial I}{\partial t} = 0 \quad \text{soit} \quad \nabla I^T(U) \cdot \dot{U} + I_t(U) = 0$$

$U = (u, v)^T$ coordonnées pixel du point

$I(U, t)$ est l'intensité de l'image en U à t

$\nabla I = (I_u, I_v)^T$ avec $I_u = \frac{\partial I}{\partial u}$ et $I_v = \frac{\partial I}{\partial v}$, ∇I est le gradient spatial

→ composante du flot optique selon le gradient spatial de l'intensité.

Voisinage [vois] du point (u, v) → \dot{U} est constant sur ce voisinage.

\dot{U} est déterminé en minimisant la fonctionnelle :

$$\sum_{U \in [\text{vois}]} [w^2(U) \cdot (\nabla I^T(U) \cdot \dot{U} + I_t(U))]^2$$

$$W^2 A \dot{U} = W^2 b$$

$$A = \underbrace{\begin{pmatrix} \nabla I(U_1) \\ \vdots \\ \nabla I(U_n) \end{pmatrix}}_{\text{gradients spatiaux}}, \quad W = \underbrace{\begin{pmatrix} w^2(U_1) & & 0 \\ & \ddots & \\ 0 & & w^2(U_n) \end{pmatrix}}_{\text{poids du voisinage}} \quad \text{et} \quad b = \underbrace{-(I_t(U_1) \cdots I_t(U_n))}_{\text{gradients temporels}}$$

$$\text{mmc} \rightarrow \dot{U} = (A^T W^2 A)^{-1} A^T W^2 b$$

$$\text{avec } A^T W^2 A = \begin{pmatrix} \sum w^2 I_u^2 & \sum w^2 I_u I_v \\ \sum w^2 I_u I_v & \sum w^2 I_v^2 \end{pmatrix} \text{ et } A^T W^2 b = - \left(\sum w^2 I_u I_t \right)$$

Flot optique défini si $A^T W^2 A$ est inversible.

Problèmes à la frontière des objets :
discontinuité du flot optique.

Applications

détection d'objets en mouvement,
segmentation de scène.

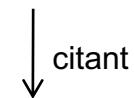
asservissement visuel

recouvrement des coordonnées 3D de points

Sources

John Leonard Barron, David J. Fleet &
Steven Simon Beauchemin , 1994.

Performance of optical flow techniques.
International Journal of Computer Vision,
Kluwer Academic Publishers, , 12(1), pp.
43-77

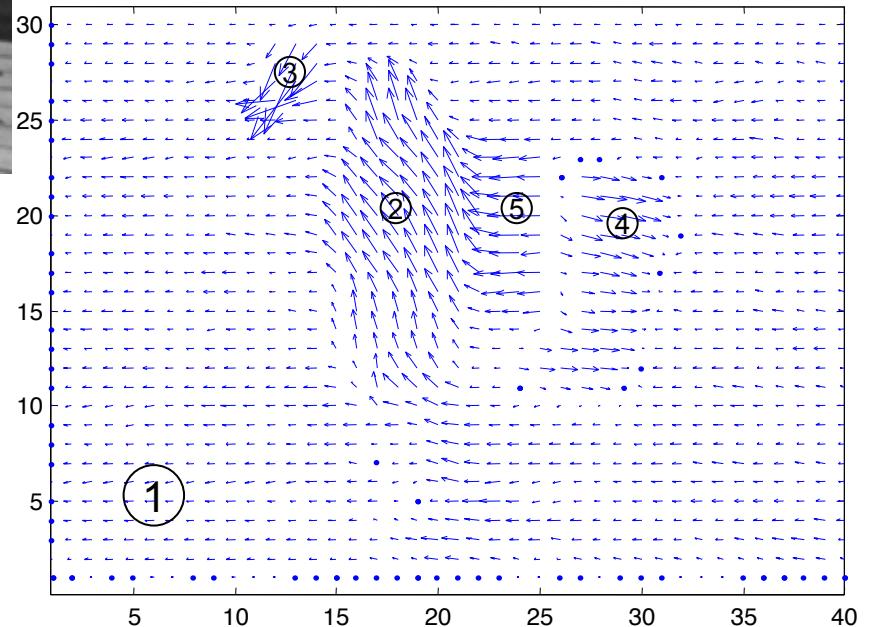


Bruce D. Lucas & Takeo Kanade , 1981. *An Iterative Image Registration Technique with an Application to Stereo Vision*. Proceedings of the 1981 DARPA Image Understanding Workshop, , pp. 121-130

Flot Optique Exemple



1. Mouvement global
2. Les deux petites filles
3. Le ballon
4. La petite fille tournant autour du mat
5. Le petit garçon qui marche



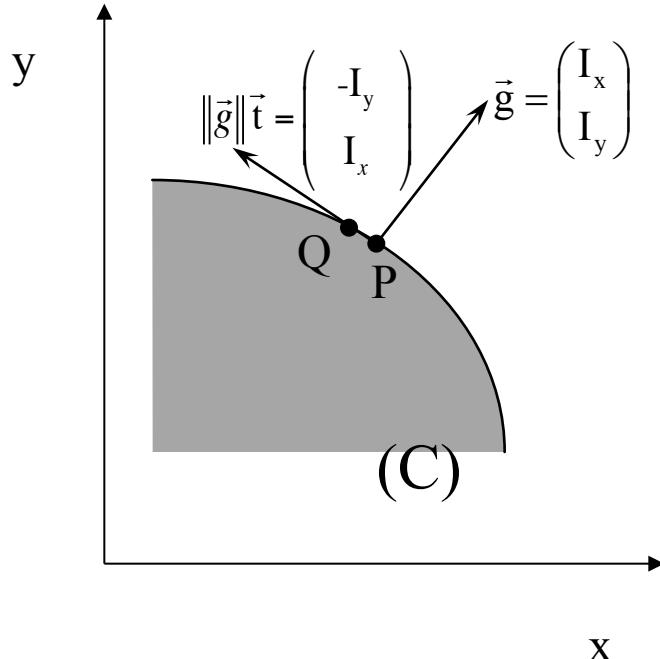
Recherche de points d'intérêts dans une image : Différentes approches

- Image $I(x,y)$ traitée par un opérateur gradient (Intensité, niveau de gris):
 - recherche des points de plus forte courbure sur les contours
- Surface $z = f(x,y)$
 - recherche des points stationnaires
 - recherche des points de plus forte courbure
- Image $I(x,y)$
 - recherche de points pour lesquels l'intensité moyenne sur une fenêtre les entourant varie fortement pour un petit déplacement*

$$E_{x,y} = \sum_{u,v} w_{u,v} |I_{x+u,y+v} - I_{x,y}|^2$$

* principe du détecteur de Moravec

Courbure de $I(x,y)$ (1/2)



$$I(x,y) \text{ image d'intensité}, \frac{\partial I}{\partial x} = I_x, \frac{\partial I}{\partial y} = I_y$$

(C) courbe iso-intensité: $I(x,y)=c^{\text{te}} = c$

$$P \begin{pmatrix} x \\ y \end{pmatrix} \text{ et } Q \begin{pmatrix} x+\delta x \\ y+\delta y \end{pmatrix} \in (C)$$

$$I(Q)-I(P)=0=\delta x I_x + \delta y I_y + \text{termes ordre sup}$$

$$\text{vrai pour tout vecteur } h = \begin{pmatrix} \delta x \\ \delta y \end{pmatrix} \in (C)$$

$$\text{donc } \vec{g} = \begin{pmatrix} I_x \\ I_y \end{pmatrix}_P \perp (C) \text{ en } P$$

$$\vec{t} \text{ vecteur unitaire tangent à (C): } \|\vec{g}\| \vec{t} = \begin{pmatrix} -I_y \\ I_x \end{pmatrix}$$

$$\vec{t} = \left(\frac{dx}{ds}, \frac{dy}{ds} \right)^t \quad s : \text{abscisse curviligne}$$

$$\vec{n} \text{ vecteur unitaire normal à (C): } \frac{\vec{g}}{\|\vec{g}\|}$$

$$\text{Courbure} = \left\| \frac{d\vec{t}}{ds} \right\|$$

Courbure de $I(x,y)$ (2/2)

- S : abscisse curviligne le long de (C)

- Courbure k avec $k=1/\rho$, ρ = rayon de courbure

$$\vec{g} \cdot \vec{t} = 0 \Rightarrow \frac{d(\vec{g} \cdot \vec{t})}{ds} = \frac{d\vec{g}}{ds} \cdot \vec{t} + \vec{g} \cdot \frac{d\vec{t}}{ds} = 0 \text{ avec } \vec{g} = \begin{pmatrix} I_x \\ I_y \end{pmatrix} \text{ & } \vec{t} = \begin{pmatrix} \frac{dx}{ds} \\ \frac{dy}{ds} \end{pmatrix}$$

$$\frac{d\vec{t}}{ds} = k\vec{n} = k \frac{\vec{g}}{\|\vec{g}\|}$$

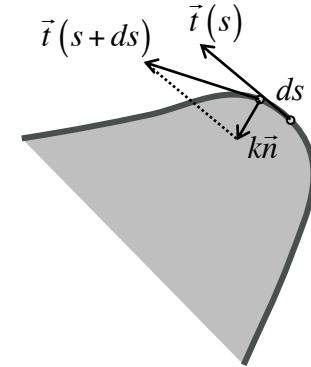
$$\frac{d\vec{g}}{ds} = \frac{\partial \vec{g}}{\partial x} \frac{dx}{ds} + \frac{\partial \vec{g}}{\partial y} \frac{dy}{ds} = H\vec{t} \text{ avec } H = \begin{bmatrix} \frac{\partial I_x}{\partial x} & \frac{\partial I_x}{\partial y} \\ \frac{\partial I_y}{\partial x} & \frac{\partial I_y}{\partial y} \end{bmatrix} = \underbrace{\begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}}_{\text{Hessienne} = \text{Jacobienne du gradient}} \text{ avec } I_{xx} = \frac{\partial^2 I}{\partial x^2}, I_{yy} = \frac{\partial^2 I}{\partial y^2}, I_{xy} = \frac{\partial^2 I}{\partial x \partial y}$$

$$k\|\vec{g}\| = \frac{-\vec{t}^T H \vec{t}}{\|\vec{g}\|} \quad \|\vec{g}\| \vec{t} = \begin{pmatrix} -I_y \\ I_x \end{pmatrix}$$

$$k\|\vec{g}\| = \frac{-\begin{pmatrix} -I_y & I_x \end{pmatrix} H \begin{pmatrix} -I_y \\ I_x \end{pmatrix}}{\|\vec{g}\|^2}$$

$$K = |k\|\vec{g}\|| = (I_{xx}I_y^2 + I_{yy}I_x^2 - 2I_{xy}I_xI_y) / (I_x^2 + I_y^2)$$

K: courbure \times norme du gradient (méthode de Kitchen et Rosenfeld)



Kitchen, L. and Rosenfeld, A. (1982). Gray-level corner detection. *Pattern Recognition Letters*, 1(2):95–102.



Recherche des points caractéristiques des surfaces $z = I(x,y)$

Points stationnaires d'une fonction $f(x,y,z)=0$

$$z = I(x, y)$$

$$z(a + \partial x, b + \partial y) - z(a, b) = \Delta z$$

$$= \partial x I_x + \partial y I_y + \frac{1}{2}(\partial x^2 I_{xx} + 2\partial x \partial y I_{xy} + \partial y^2 I_{yy}) + \dots$$

point stationnaire: $I_x = I_y = 0$

$$\Delta z = \frac{1}{2} h^t H h \quad h = \begin{pmatrix} \delta x \\ \delta y \end{pmatrix}, \quad H = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix}$$

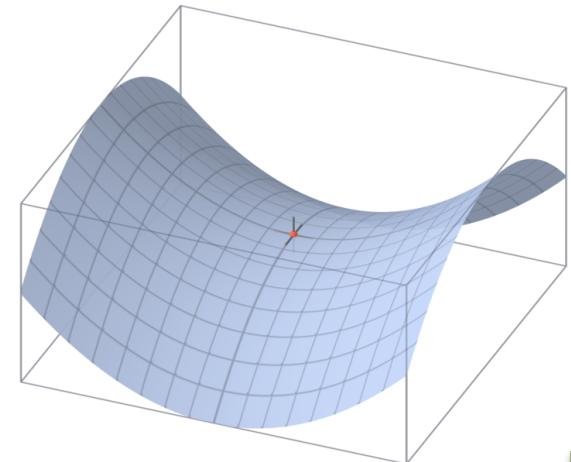
les valeurs propres sont racines de $\lambda^2 - \text{Trace}(H)\lambda + \det(H) = 0$

$\Delta z > 0$ si $\det(H) > 0$ et (λ_1 et $\lambda_2 > 0$) forme ellipsoidale, minimum

$\Delta z < 0$ si $\det(H) > 0$ et (λ_1 et $\lambda_2 < 0$) forme ellipsoidale, maximum

$\det(H) < 0$, λ_1 et λ_2 signes opposés, forme hyperbolique, min/max

$\det(H) = 0$, λ une valeur propre ≈ 0 , forme parabolique, variations nulles dans une direction



SVD

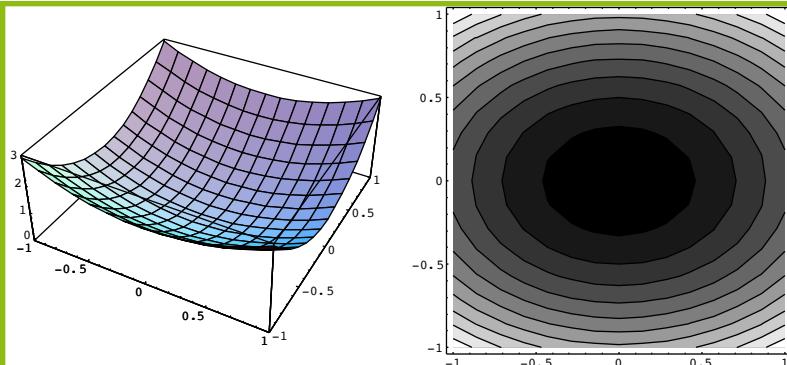
les valeurs propres λ_1 et λ_2 sont des mesures des courbures principales au point considéré.

$\text{Trace}(H)$ est donc une mesure de la courbure moyenne

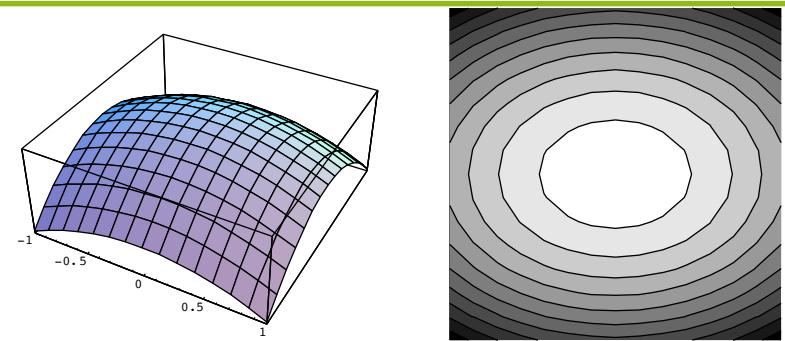
$\text{Det}(H)$ est le produit des courbures (dite courbure gaussienne)

(méthode de Beaudet page 102)

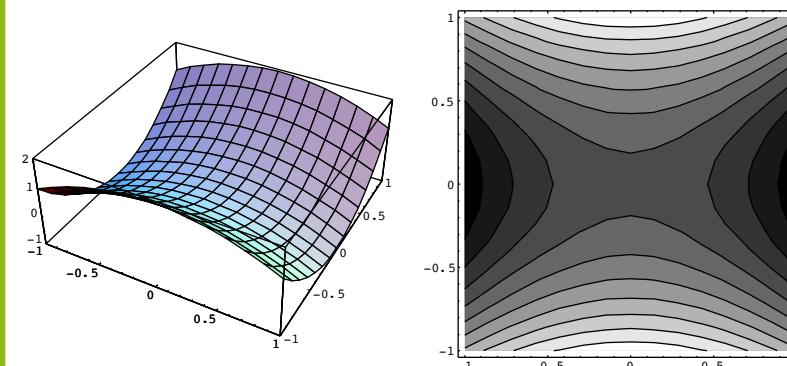
Points stationnaires d'une fonction $f(x,y,z)=0$



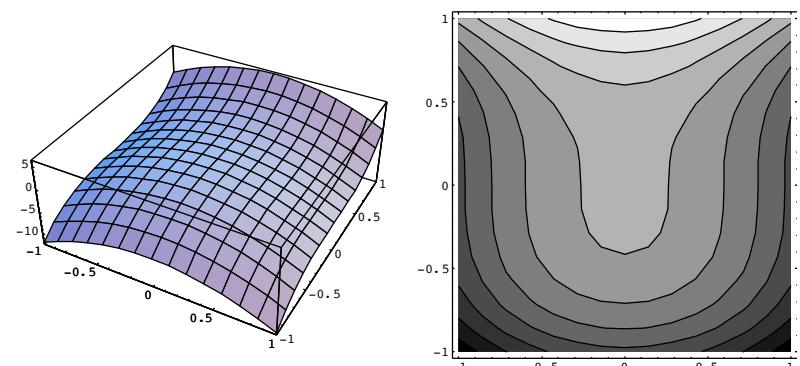
$\det(H) > 0$ et $I_{xx}, I_{yy} > 0$
forme elliptique minimum



$\det(H) > 0$ et $I_{xx}, I_{yy} < 0$
forme elliptique maximum



$\det(H) < 0$
forme hyperbolique, min/max



$\det(H) = 0$
forme parabolique

Courbure gaussienne en un point quelconque de la surface de l'image

$f(x, y, z) = 0$ tel que $z = I(x, y)$ $f \equiv I(x, y) - z$

$$f(a+h) - f(a) = h^t g + \dots = 0 \text{ où } a = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \text{ et } h = \begin{pmatrix} \partial x \\ \partial y \\ \partial z \end{pmatrix}$$

vecteur gradient $g = \begin{pmatrix} I_x \\ I_y \\ -1 \end{pmatrix}$ tel que $h^t g = 0$ (propriété du vecteur gradient)

$$n \text{ vecteur unitaire normal à la surface: } n = \frac{1}{\sqrt{I_x^2 + I_y^2 + 1}} \begin{pmatrix} I_x \\ I_y \\ -1 \end{pmatrix}$$

La composante dn parallèle à la surface en fonction d'un petit déplacement h s'écrit alors (en supposant que $\|g\|$ est quasi constant):

$$dn = \frac{1}{\sqrt{I_x^2 + I_y^2 + 1}} H h \text{ avec } H = \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix}$$

La courbure gausienne (produit des deux courbures principales) est alors donnée par :

$$G = \frac{\det(H)}{\|\bar{g}\|^2} = \frac{I_{xx}I_{yy} - I_{xy}^2}{I_x^2 + I_y^2 + 1} \quad (\text{méthode de Dreschler et Nagel})$$

Points d'intérêts

- recherche de points pour lesquels l'intensité moyenne sur une fenêtre les entourant varie fortement pour un petit déplacement

$$E_{a,b} = \sum_{u,v} w_{u,v} |I_{a+u,b+v} - I_{a,b}|^2$$

$$I(a+h) - I(a) = h^t g + \dots \quad a = \begin{pmatrix} x \\ y \end{pmatrix}, h = \begin{pmatrix} \delta x \\ \delta y \end{pmatrix}, g = \begin{pmatrix} I_x \\ I_y \end{pmatrix}$$

$$(I(a+h) - I(a))^2 = h^t M h \quad M = \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix}$$

En fait l'image est lissée par une fonction , par exemple du type $w(u, v) = \frac{1}{\sigma^2} e^{-(u^2+v^2)/2}$

soit \hat{I} l'image lissée

$$M \text{ devient alors égale à } \hat{C} = \begin{pmatrix} \hat{I}_x^2 & \hat{I}_x \hat{I}_y \\ \hat{I}_x \hat{I}_y & \hat{I}_y^2 \end{pmatrix}$$

Trace(\hat{C}) est une mesure de la courbure moyenne

Det(\hat{C}) est une mesure de la 'courbure gaussienne' (le produit des courbures principales)

Noble, Plessey utilisent l'opérateur $C_p = \text{Trace}(\hat{C}) / \text{Det}(\hat{C})$

Harris utilise l'opérateur $R = \text{Det}(\hat{C}) - k \text{Trace}^2(\hat{C})$, avec selon les auteurs $k \in [0.04 \cdots 0.15]$

Exemple : détecteur de coins de Harris (1/2)

- Soit un voisinage (u, v) l'image I décalée de (x, y) :

$$S(x, y) = \sum_u \sum_v \underbrace{w(u, v)}_{\text{weight}} \underbrace{\left(I(u, v) - I(u + x, v + y) \right)^2}_{\text{Sum of Square Differences}}$$

$w(u, v)$ = gaussienne par exemple

$$I(u + x, v + y) \approx I(u, v) + I_x(u, v)x + I_y(u, v)y + \dots$$

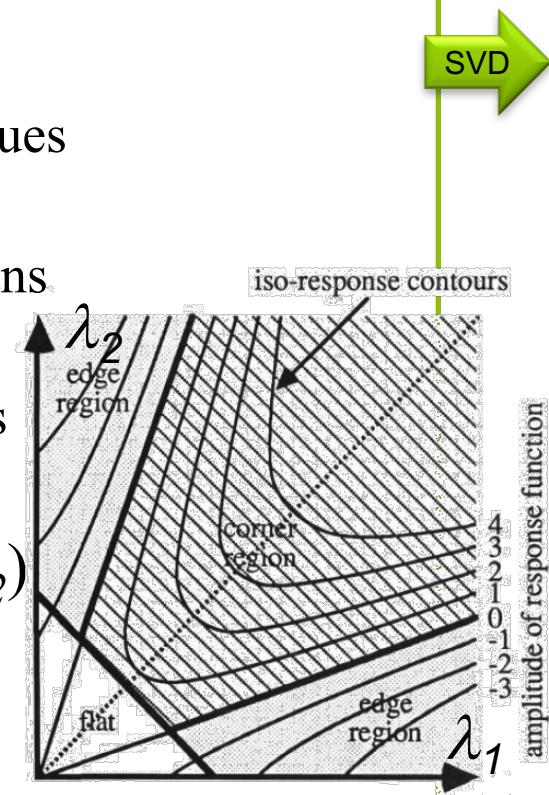
$$S(x, y) \approx \sum_u \sum_v w(u, v) \left(I_x(u, v)x + I_y(u, v)y \right)^2$$

$$\approx \begin{pmatrix} x & y \end{pmatrix} A \begin{pmatrix} x \\ y \end{pmatrix}$$

$$A = \sum_u \sum_v w(u, v) \underbrace{\begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix}}_{\text{Hessian Matrix}} = \begin{pmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{pmatrix}$$

Exemple : détecteur de coins de Harris (2/2)

- Un « coin » est caractérisé par une grande variation de S dans toutes les directions
- Soit λ_1 & λ_2 les valeurs propres de A :
 - $\lambda_1 \approx 0$ & $\lambda_2 \approx 0$: le pixel (x,y) n'a pas caractéristiques d'intérêt (au sens des coins) : zone homogène
 - $\lambda_1 \approx 0$ & $\lambda_2 \gg 0$: indique un contour (variation dans une seule direction)
 - $\lambda_1 \gg 0$ & $\lambda_2 \gg 0$: indique un coin (variation dans les deux directions)
- Mesure de Harris (évite le calcul de λ_1 & λ_2)
$$M_c = \lambda_1 \lambda_2 - \kappa (\lambda_1 + \lambda_2)^2 = \det(A) - \kappa (\text{trace}(A))^2$$
avec $\kappa \in [0.04 \dots 0.15]$



Harris, C. & Stephens, M. - *A combined corner and edge detector*. Proceedings of the 4th Alvey Vision Conference (August 1988)

Exemple Feature points (SIFT)

- SIFT : Scale Invariant Feature Transform
 - But : détection de points d'intérêts (feature points) aisément reconnaissables dans d'autres images du même objet.
 - Détection de points :
 - Min et Max de Différences de Gaussiennes (DoG) dans un espace d'échelle
 - Mise en correspondance de feature sur points :
 - Distance euclidienne entre vecteurs de descripteurs (typiquement 128 composant)
 - Invariance par
 - Translation : la position n'intervient pas dans le matching
 - Rotation : les feature points sont orientés suivant direction du gradient
 - Échelle : l'utilisation d'un espace d'échelle (Scale Space) donne un facteur d'échelle aux feature points détectés
 - Invariance *partielle* à
 - Eclairage
 - Transformation affine ou projection



Espace d'échelle

- Gaussian Scale Space

$$G(x; \sigma) \stackrel{\Delta}{=} (g_\sigma \otimes I)(x)$$

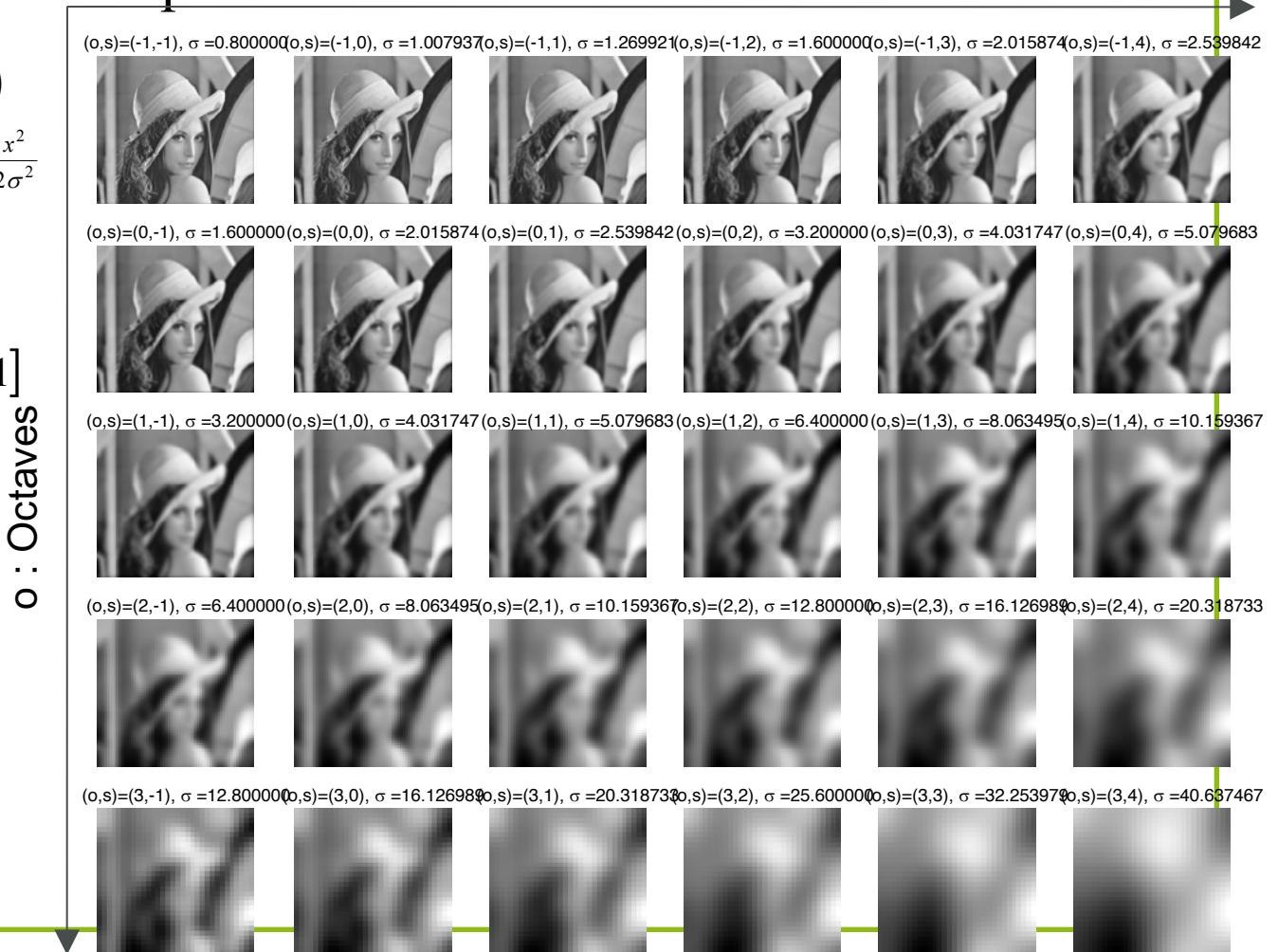
avec $g_\sigma = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$

$$\sigma(o, s) = \sigma_0 2^{o+\frac{s}{S}}$$

$$o \in o_{\min} + [0, \dots, O-1]$$

$$s \in [0, \dots, S-1]$$

s : Sous-niveaux dans l'octave



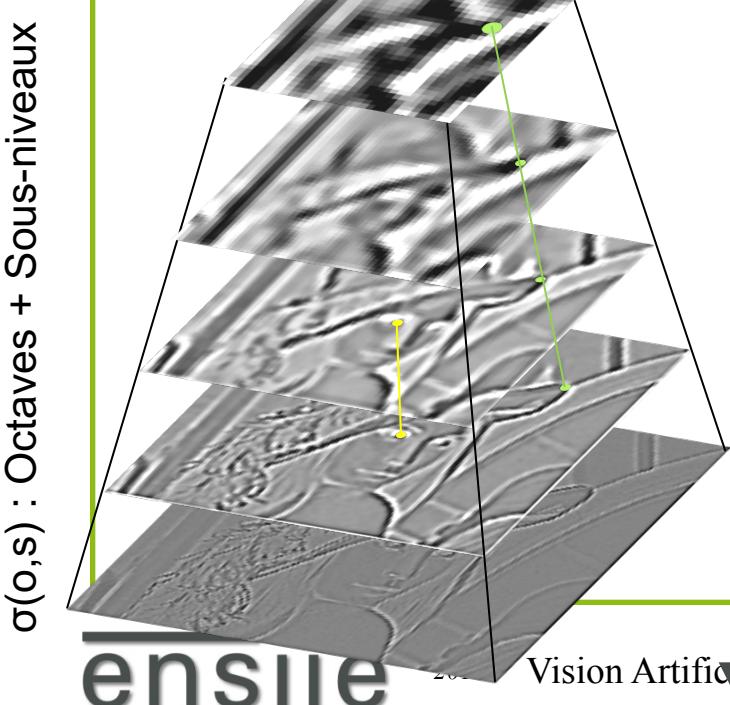
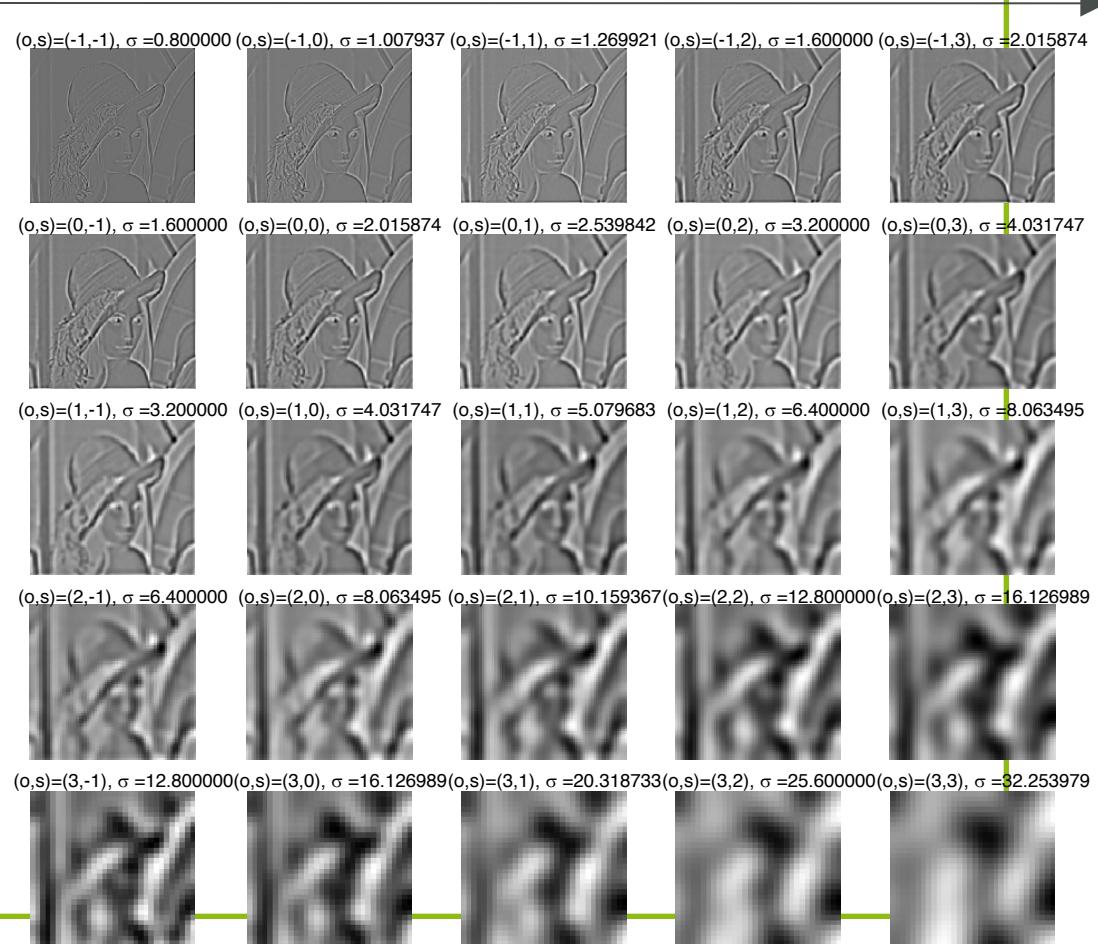
Espace d'échelle différentiel

Exemple

- DoG Scale Space
 - Stabilité

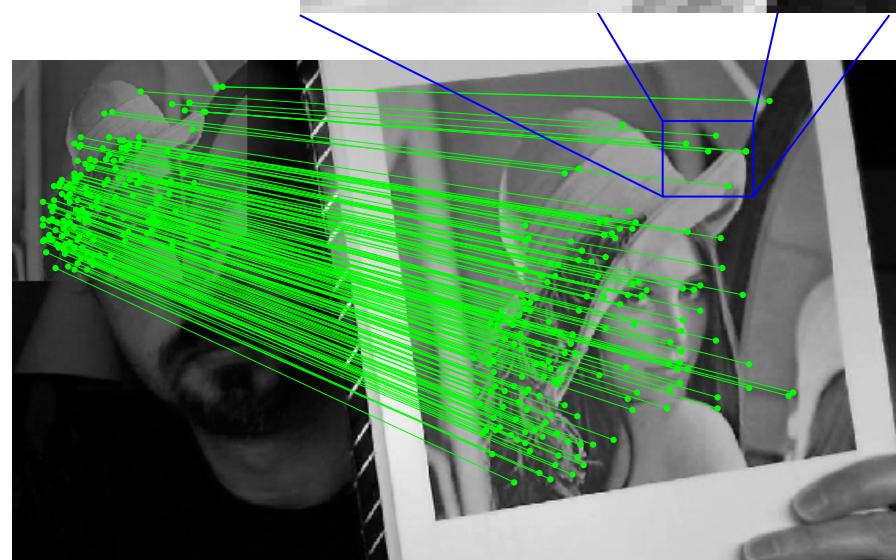
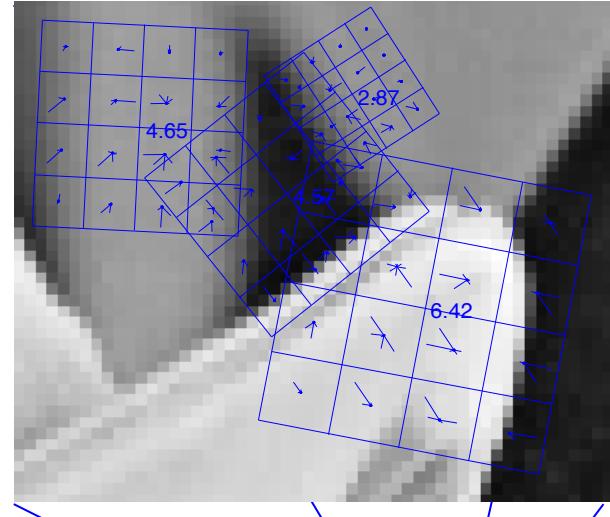
$$D(x; \sigma(o, s)) = G(x, \sigma(o, s+1)) - G(x, \sigma(o, s))$$

s : sous-niveaux dans l'octave



SIFT : descripteurs & matching

- Descripteurs
 - Histogramme locaux des orientations de gradient dans le voisinage à l'échelle du point détecté suivant son orientation
- $128 = N_p^2 N_o \begin{cases} \bullet N_p = 4 \text{ taille du découpage du voisinage} \\ \bullet N_o = 8 \text{ orientations pour chaque histo.} \end{cases}$
- Matching
 - Entre points
 - Distance euclidienne entre les descripteurs
 - Recherche du meilleur candidat dans un k-d tree
 - Entre ensembles de points :
 - Ajustement au sens des mmc de l'**homographie** correspondante (si nb pts > 3)



Détection de points d'intérêt: références

- P. Beaudet: «Rotationnaly invariant image operators». Proceedings 4th Int. Joint conf on pattern recognition. Tokyo, pp579-583, 1978
- V. Gouet, P. Montesinos, D. Pelé, R. Deriche: «Evaluation des détecteurs de points d'intérêt pour les couleurs». Rapport de recherche, EMA/LGI2P, INRIA, CCETT, France, 2000.
- C. Harris, M. Stephens: «A combined corner and edge detectors». Proc. 4th Alvey Vision Conf., Manchester, UK, pp189-192, 1988.
- J.M. Jolion, S. Bres: «Multiresolution Contrast based detection of interest points». Rapport de recherche RR-98-02 , INSA Lyon, 1998.
- H.P. Moravec: «Towards automatic visual obstacle avoidance». Proc. Int. Joint Conf. on Artificial Intelligence, Cambridge, USA, pp584-....., 1977.
- S.M. Smith, J.M. Brady: «A new approach to low level image processing». Technical report, Robotics research group, Department of engineering science, Université d'Oxford, UK, 1995.
- R. Deriche, G. Giraudon: «A computational approach for corner and vertex detection». Int. Journal of computer vision, 10:2, pp 101-124, 1993.
- D. G. Lowe: « Object Recognition from Local Scale-Invariant Features ». Seventh IEEE International Conference on Computer Vision (ICCV'99), pp 1150, 1999
- D. G. Lowe. « Distinctive image features from scale-invariant keypoints ». International Journal of Computer Vision . Springer Netherlands, Vol 60 , n° 2, 91-110 (November 2004)



Modélisation
Reconnaisances de formes
Indexation
Localisation d'un objet

Modélisation de l'image d'un objet

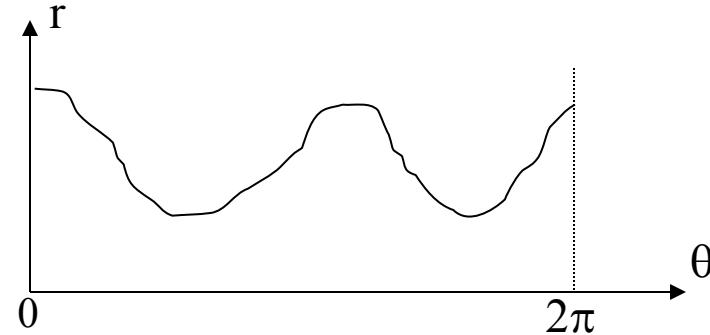
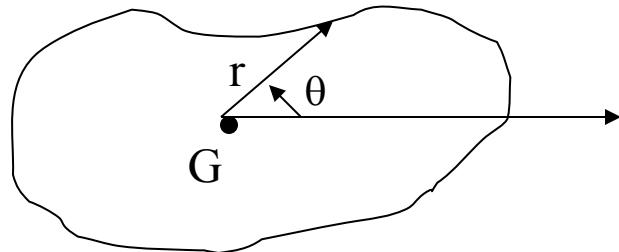
- Objectif:
 - réduire la quantité d'informations à traiter pour permettre le stockage, le traitement, la reconnaissance de forme, l'indexation, l'appariement.
- Modélisation
 - paramétrique
 - modèle = vecteur de paramètres descripteurs de contours, surfaces
 - structurelle
 - modèle = une chaîne de caractères, un arbre ou un graphe
 - ensemble de caractéristiques locales liées par des relations géométriques et/ou topologiques
- Modélisation \neq Transformation
 - transformation \rightarrow reconstruire la forme
 - applications: filtrage, compression pour stockage et transmissions vidéo)
 - modélisation \rightarrow reconnaissance de formes pré-cataloguées

Descripteurs de l'image d'objets

- Objectifs
 - indépendants de la position de l'objet dans l'image (invariance en translation et en rotation)
 - éventuellement indépendant de l'échelle (invariance par homothétie)

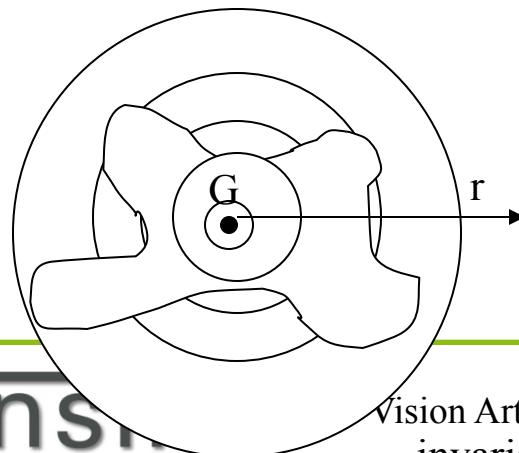
Descripteurs d'objets

représentation polaire

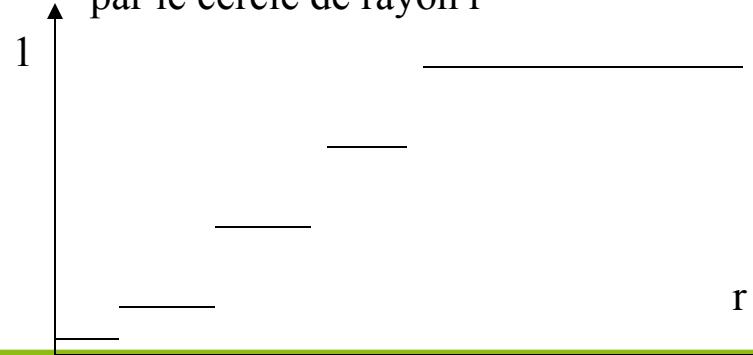


$r(\theta)$ ou coefficients de Fourier d'amplitude significative
invariant en translation
invariant en rotation si les modules des coef. de Fourier
orientation contenue dans la phase des coef. de Fourier

histogramme des cercles de rayon croissant



proportion de la surface de l'objet couverte
par le cercle de rayon r



Codage des contours (codage de Freeman)

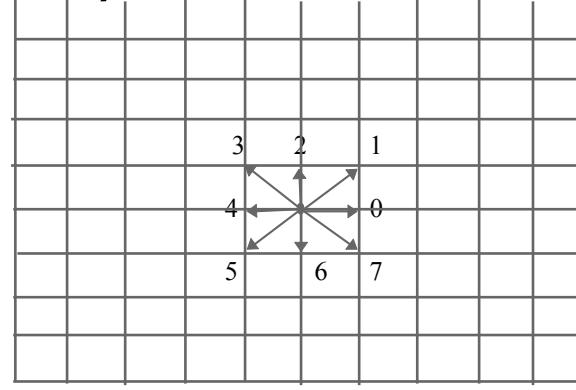
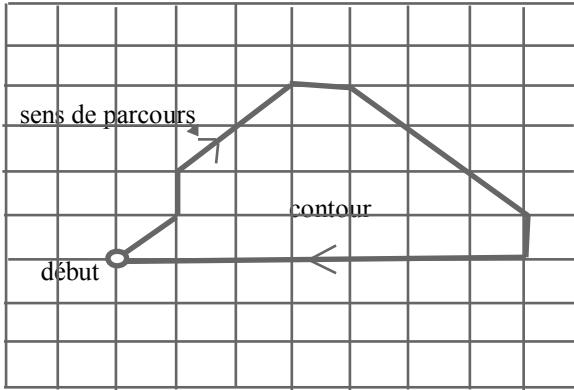
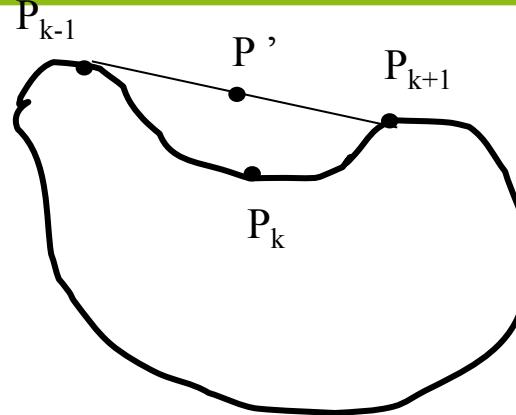
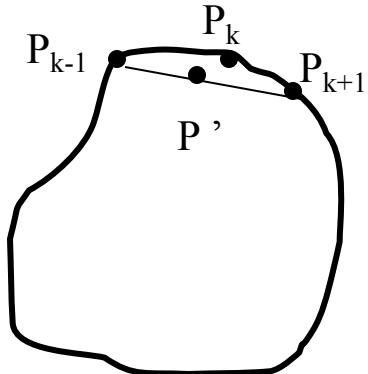


Figure 1.12. Représentation d'une forme par une chaîne de caractères

Dans la figure 1.12, chaque nœud de la grille représente un pixel. A partir d'un pixel donné, les huit pixels voisins sont dans des directions codées de 0 à 7, appelées codage de Freeman. Le codage de Freeman constitue un alphabet ($\{0, 1, 2, 3, 4, 5, 6, 7\}$) sur lequel un contour est représenté par une chaîne de caractères ou phrase. Le contour de la figure est représenté par la chaîne de caractères : 1211077764444444

Codage des contours (concavité-convexité ou courbure)

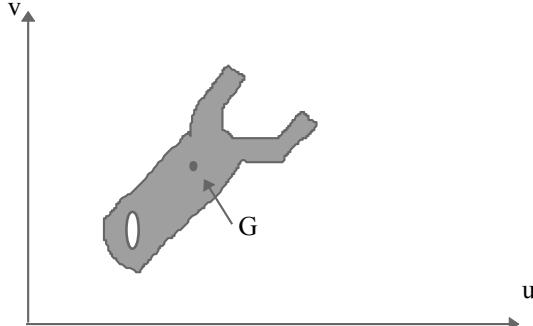


- Le contour est représenté par $\{P_k\}, k \in [1 \cdots N]$
- Codage concavité/convexité : contour = suite binaire
 - Le contour est dit convexe en P_k (codage par 0) si
$$P' \begin{cases} x' = (x_{k+1} + x_{k-1})/2 \\ y' = (y_{k+1} + y_{k-1})/2 \end{cases}$$
est tel que $P' \in$ objet
 - Le contour est dit concave en P_k (codage par 1) si $P' \notin$ objet
- Codage de la courbure en P_k : contour = suite de nombres réels $\in \mathbb{R}$ (discrétisé)

Caractéristiques globales

La forme peut être modélisée à partir de paramètres globaux de l'image de l'objet :

- A : aire,
- P : périmètre
- A/P^2 : compacité
- δ : diamètre du cercle circonscrit, δ/P
- nombre de trous : nt
- moment d'ordre 1 : (u_G, v_G)
- moments centrés d'ordres 2 : I_{uu}, I_{vv}



Exemple de forme

(Les moments sont calculés en pondérant les pixels par leur éclairement. I_{uu}, I_{vv} représentent ici les valeurs propres de la matrice des moments, i.e. les moments selon les axes principaux).

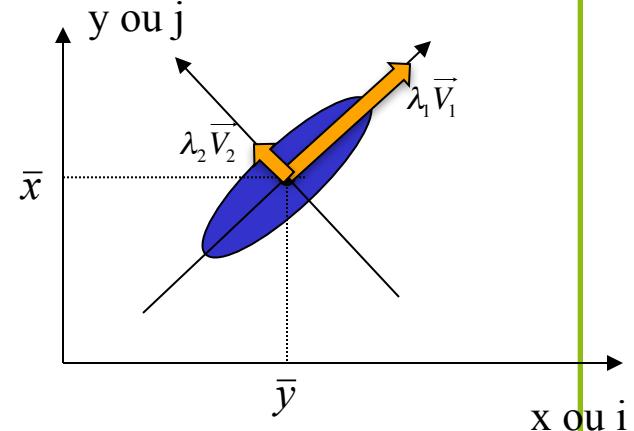
Exemple de modèle paramétrique de la forme : vecteur $(A/P^2, \delta/P, nt, I_{uu}, I_{vv})$.

Les paramètres utilisés ici sont invariants en translation et rotation.

Moments et axes principaux « d'inertie »

- Moment sur une région Ω de l'image I : $m_{pq}(I) = \sum_{x,y \in \Omega} x^p y^q, \forall (p,q) \in \mathbb{N}^2$
- $m_{00} = P$ le nombre de pixels de la région Ω
- Barycentre (\bar{x}, \bar{y}) de la région ou de l'objet : $(\bar{x}, \bar{y}) = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$

$$\bar{x} = \frac{\sum_{x,y \in \Omega} x}{P}, \bar{y} = \frac{\sum_{x,y \in \Omega} y}{P}$$



- Moments centrés : $\mu_{pq}(I) = \sum_{x,y \in \Omega} (x - \bar{x})^p (y - \bar{y})^q, \forall (p,q) \in \mathbb{N}^2$
- Moments centrés d'ordre 2 :

$$I_{xx} = \frac{\sum_{x,y \in \Omega} (x - \bar{x})^2}{P} = \frac{\mu_{20}}{\mu_{00}}, I_{yy} = \frac{\sum_{x,y \in \Omega} (y - \bar{y})^2}{P} = \frac{\mu_{02}}{\mu_{00}}, I_{xy} = \frac{\sum_{x,y \in \Omega} (x - \bar{x})(y - \bar{y})}{P} = \frac{\mu_{11}}{\mu_{00}}$$

- Permettent de retrouver les axes principaux d'inertie de l'objet

ou de la région et donc de retrouver son orientation

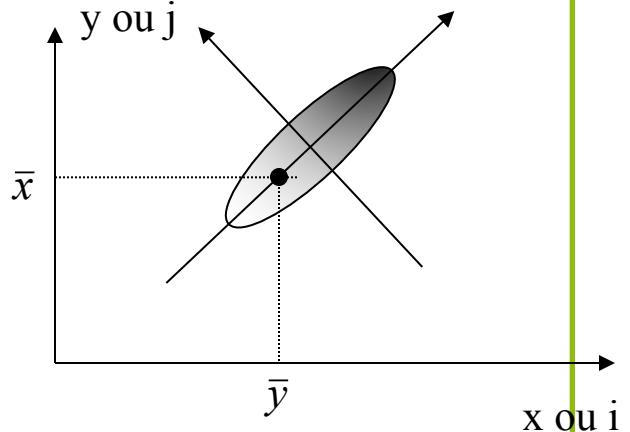
$$svd \begin{pmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{pmatrix} \Rightarrow \underbrace{\begin{pmatrix} & \\ & \end{pmatrix}}_{U_{2 \times 2}} \underbrace{\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}}_{S_{2 \times 2}} \underbrace{\begin{pmatrix} V_1 \\ V_2 \end{pmatrix}}_{V_{2 \times 2}^T}$$

Moments et axes principaux « d'inertie » pondérés par $I(x,y)$

- Moment sur une région Ω de l'image I : $m_{pq}(I) = \sum_{x,y \in \Omega} x^p y^q I(x,y), \forall (p,q) \in \mathbb{N}^2$

- Barycentre (\bar{x}, \bar{y}) de la région ou de l'objet : $(\bar{x}, \bar{y}) = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$

$$\bar{x} = \frac{\sum_{x,y \in \Omega} x I(x,y)}{\sum_{x,y \in \Omega} I(x,y)}, \bar{y} = \frac{\sum_{x,y \in \Omega} y I(x,y)}{\sum_{x,y \in \Omega} I(x,y)}$$



- Moments centrés : $\mu_{pq}(I) = \sum_{x,y \in \Omega} (x - \bar{x})^p (y - \bar{y})^q I(x,y), \forall (p,q) \in \mathbb{N}^2$
- Moments centrés d'ordre 2 :

$$I_{xx} = \frac{\sum_{x,y \in \Omega} (x - \bar{x})^2 I(x,y)}{\sum_{x,y \in \Omega} I(x,y)} = \frac{\mu_{20}}{\mu_{00}}, I_{yy} = \frac{\sum_{x,y \in \Omega} (y - \bar{y})^2 I(x,y)}{\sum_{x,y \in \Omega} I(x,y)} = \frac{\mu_{02}}{\mu_{00}}, I_{xy} = \frac{\sum_{x,y \in \Omega} (x - \bar{x})(y - \bar{y})}{\sum_{x,y \in \Omega} I(x,y)}$$

- Permettent de retrouver les axes principaux d'inertie de l'objet ou de la région et donc de retrouver son orientation

Axes principaux « d'inertie »

La diagonalisation de la matrice des moments centrés permet de déterminer les axes principaux de l'objet (région) et donc de trouver son orientation

$$A = \begin{pmatrix} I_{XX} & I_{XY} \\ I_{YX} & I_{YY} \end{pmatrix} \quad A : \text{symétrique, est diagonalisable}$$

les valeurs propres λ_1 et λ_2 sont réelles

les vecteurs propres V_1 et V_2 sont indépendants et perpendiculaires

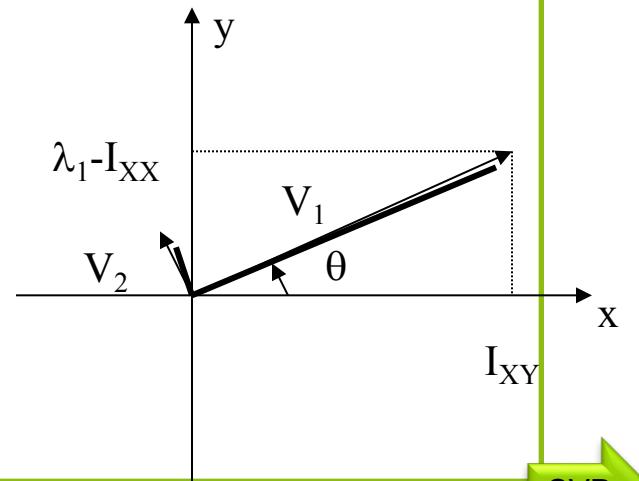
$\Rightarrow \lambda_1$ et λ_2 sont telles que : $\det[A - \lambda \text{Id}] = 0$

$$\begin{vmatrix} I_{XX} - \lambda & I_{XY} \\ I_{XY} & I_{YY} - \lambda \end{vmatrix} = 0 \quad \text{soit } \lambda^2 - (I_{XX} + I_{YY})\lambda + I_{XX} \cdot I_{YY} - I_{XY}^2 = 0$$

$$\Rightarrow V_1 \text{ et } V_2 \text{ sont tels que : } (A - \lambda \text{Id})X = 0 \quad \text{soit } V = \begin{pmatrix} x \\ y \end{pmatrix}$$

$$\begin{pmatrix} I_{XX} - \lambda & I_{XY} \\ I_{XY} & I_{YY} - \lambda \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 0 \rightarrow V_{1(\text{ou } 2)} = \begin{pmatrix} I_{XY} \\ \lambda_{1(\text{ou } 2)} - I_{XX} \end{pmatrix}$$

$$\theta = \arctg \frac{\lambda_1 - I_{XX}}{I_{XY}} \quad \text{soit} \quad \theta = \frac{1}{2} \arctg \frac{2I_{XY}}{I_{XX} - I_{YY}}$$



Autres descripteurs

- de contours
 - segments, coins, arc de cercle, courbes paramétrique
- de région ou objet
 - niveau de gris, couleur, texture
 - motifs géométriques (trous,)

Modélisation

- Modélisation paramétrique
 - A l'objet, on associe un vecteur $\mathbf{X}^T = (x_1 \dots x_i \dots x_n)$
 - les x_i sont les paramètres issus des descripteurs de contour, des caractéristiques globales (dont niveaux de gris, couleur, texture)
 - les x_i doivent présenter les conditions suivantes:
 - invariants en rotation-translation (éventuellement en échelle)
 - indépendants
 - significatifs (ACP sur la base d'apprentissage)
 - variance faible

Reconnaissance de formes

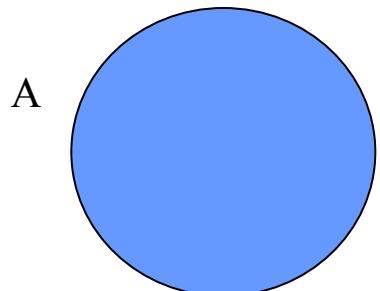
- (pattern recognition)
- classification
- apprentissage hors ligne → classes d'objets
- nouvelle image → modèle de (des) objet(s)
- reconnaissance de forme ou classification: attribution du modèle à une classe

Reconnaissance de formes à partir de modèles paramétriques

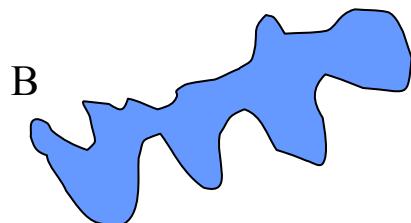
- Arbre de décision ou classification
- Plus proche voisin
 - une classe représentée par son vecteur de paramètres caractéristiques
 - X_k est le vecteur caractéristique de la classe k , $k \in (1, N)$
$$X_k^T = (x_{1}^k \quad .. \quad x_i^k \quad .. \quad x_n^k)$$
 - X est le vecteur du modèle de l'objet à reconnaître
- $d_k^2 = \|X - X_k\|^2$
distance euclidienne (au carré) entre le modèle de l'objet et l'objet représentatif de la classe k
on calcule ainsi d_k pour $k \in [1 \dots N]$
la distance minimale désigne l'objet observé en attribuant son modèle à la classe correspondante
- K plus proches voisins
- etc.....

arbre de décision

(exemple: modèle paramétrique de vecteur (P^2/A , δ/P))



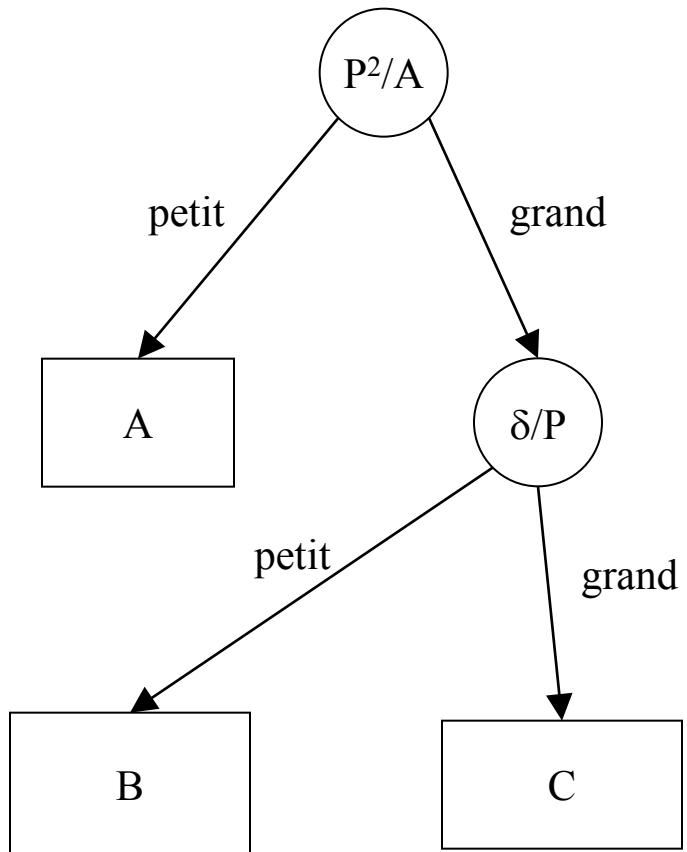
P^2/A petit
 $\delta/P = \text{moyen} (=1/\pi)$



P^2/A grand
 $\delta/P = \text{petit} (< 1/\pi)$

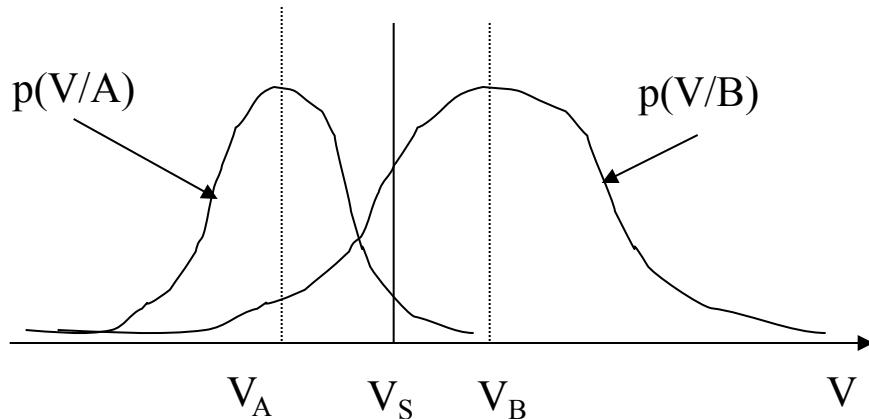


P^2/A grand
 $\delta/P = \text{grand} (> 1/\pi)$



décisions déterministes ou probabilistes (tests bayésiens par exemple)

Décision statistique: exemple



Objet représenté par un seul paramètre V

Il y a deux modèles A et B

L'analyse de V sur l'ensemble de la collection des objets fait apparaître une dispersion des valeurs caractérisées par $p(V/A)$ et $p(V/B)$, et des valeurs moyennes V_A et V_B

règle du maximum de vraisemblance (test bayésien)
si $p(V,A) > p(V,B)$ alors A est l'objet à reconnaître

mais:

$p(V,A) = p(V/A) \cdot p(A)$ où $p(A)$ est la fréquence d'apparition de l'objet A
de même $p(V,B) = p(V/B) \cdot p(B)$

Le seuil de décision V_S est caractérisé par $p(V_S/A) \cdot p(A) = p(V_S/B) \cdot p(B)$
connaissant le type de statistique (gaussienne par exemple) on en déduit la probabilité d'erreur
On peut fixer la probabilité d'erreur admissible on aura alors en général un seuil haut et un seuil bas
et entre les deux une zone de rejet

Modélisation structurelle

- Description par un ensemble de caractéristiques locales liées par des relations géométriques et/ou topologiques.
- Alphabet des descripteurs locaux:
 - mono-dimensionnels: segments, coins, arcs de cercle
 - bi-dimensionnels: trous, régions caractéristiques
- Description de la structure:
 - relations géométriques et (ou) topologiques liant des paires de descripteurs
 - voisinage, connexité, inclusion, etc..
- Structure de graphe, d'arbre ou de chaîne.
- Reconnaissance de forme en utilisant la notion de distance entre chaîne, arbre ou graphe

Modélisation structurelle

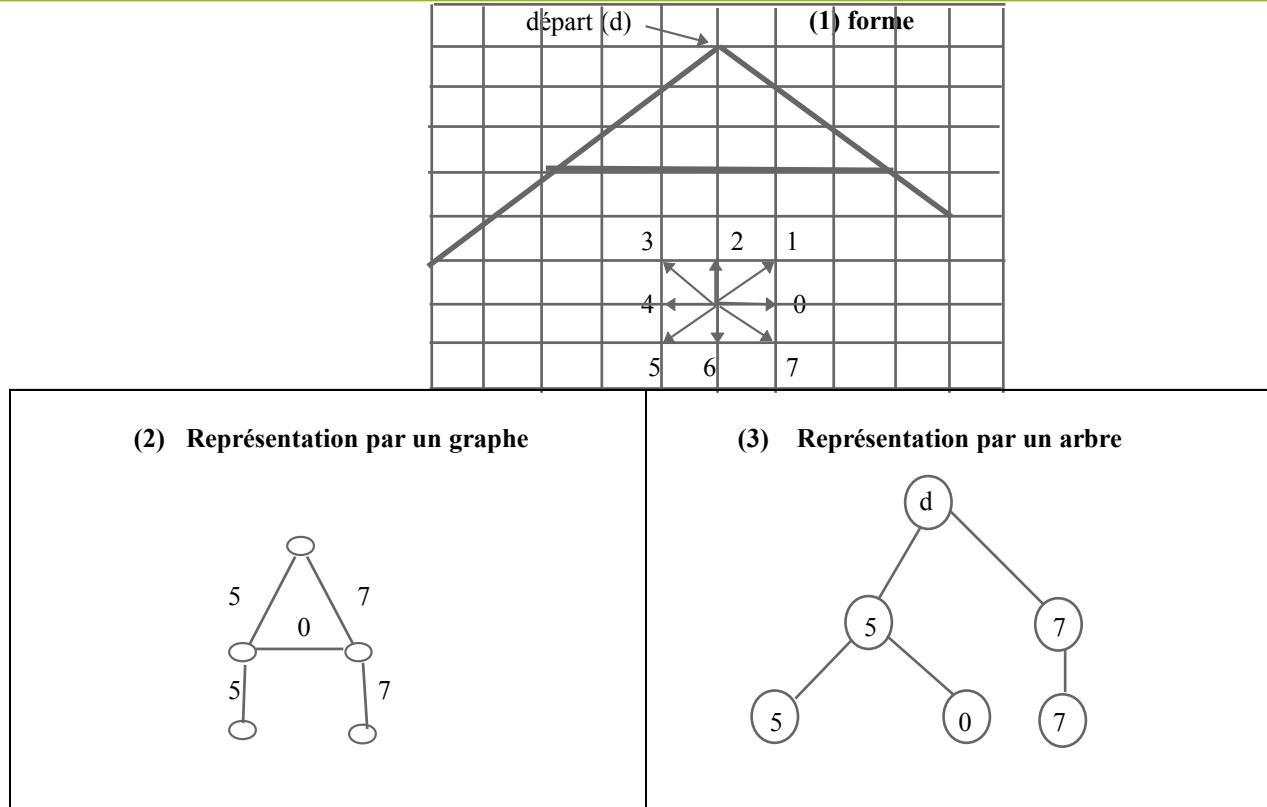
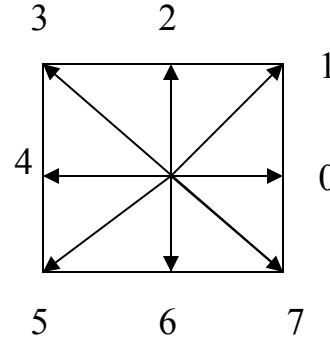
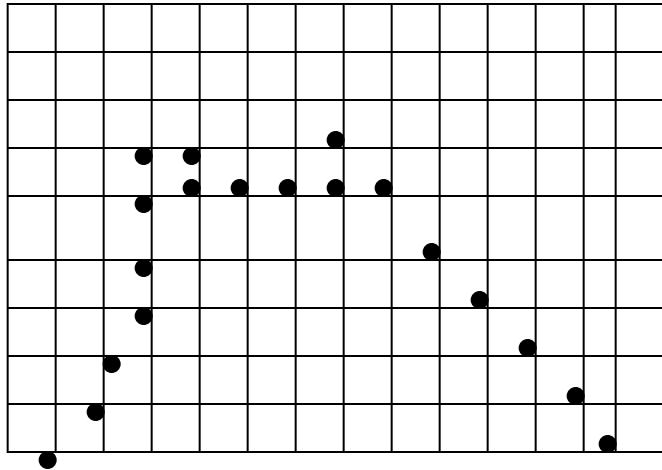


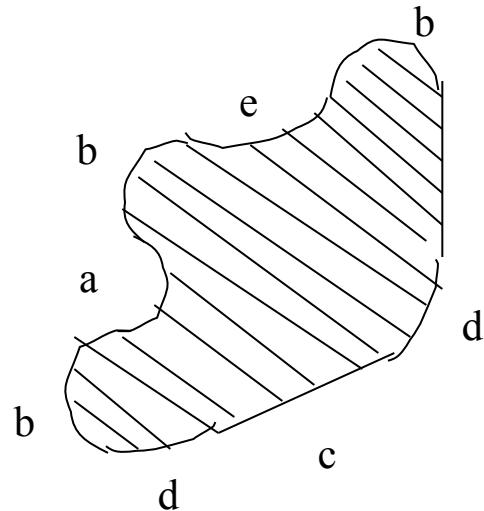
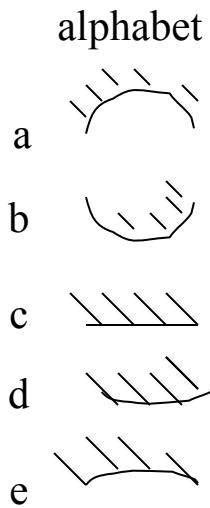
Figure 1.13. Méthodes structurelles de représentation (d'après [MIC 84]) : (1) forme, (2) représentation par un arbre, (3) représentation par un graphe

modélisation structurelle par chaîne de caractères



alphabet de Freeman
 $\{0,1,2,3,4,5,6,7\}$

‘ phrase ’ (chaîne de caractères)
représentant la courbe
[121222060002777777]



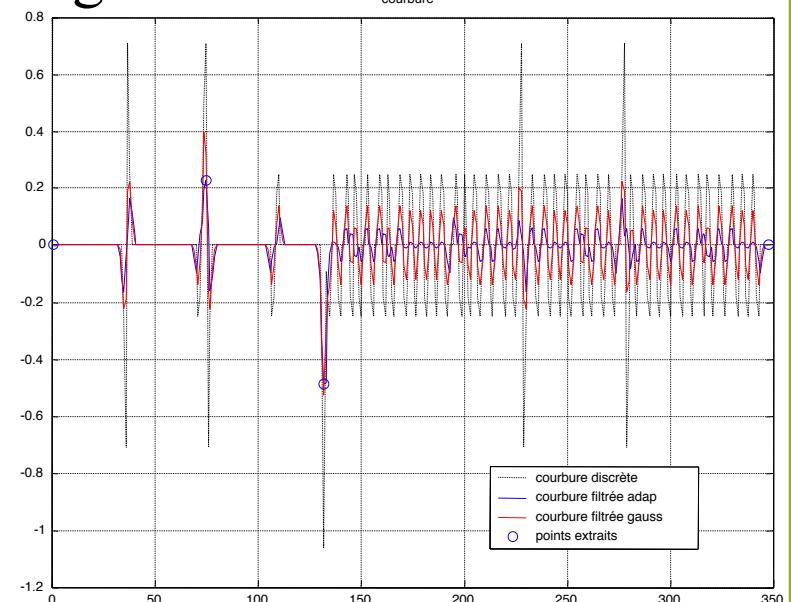
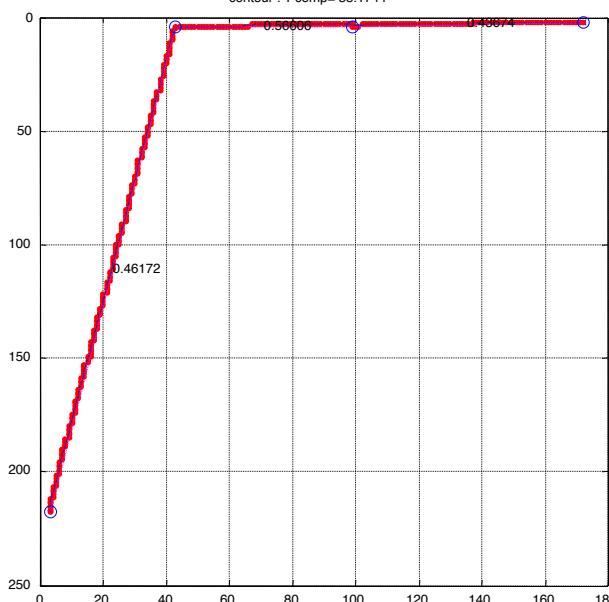
phrase (chaîne de caractère)
représentant la forme
[bcdcdbabe]

Courbure des contours

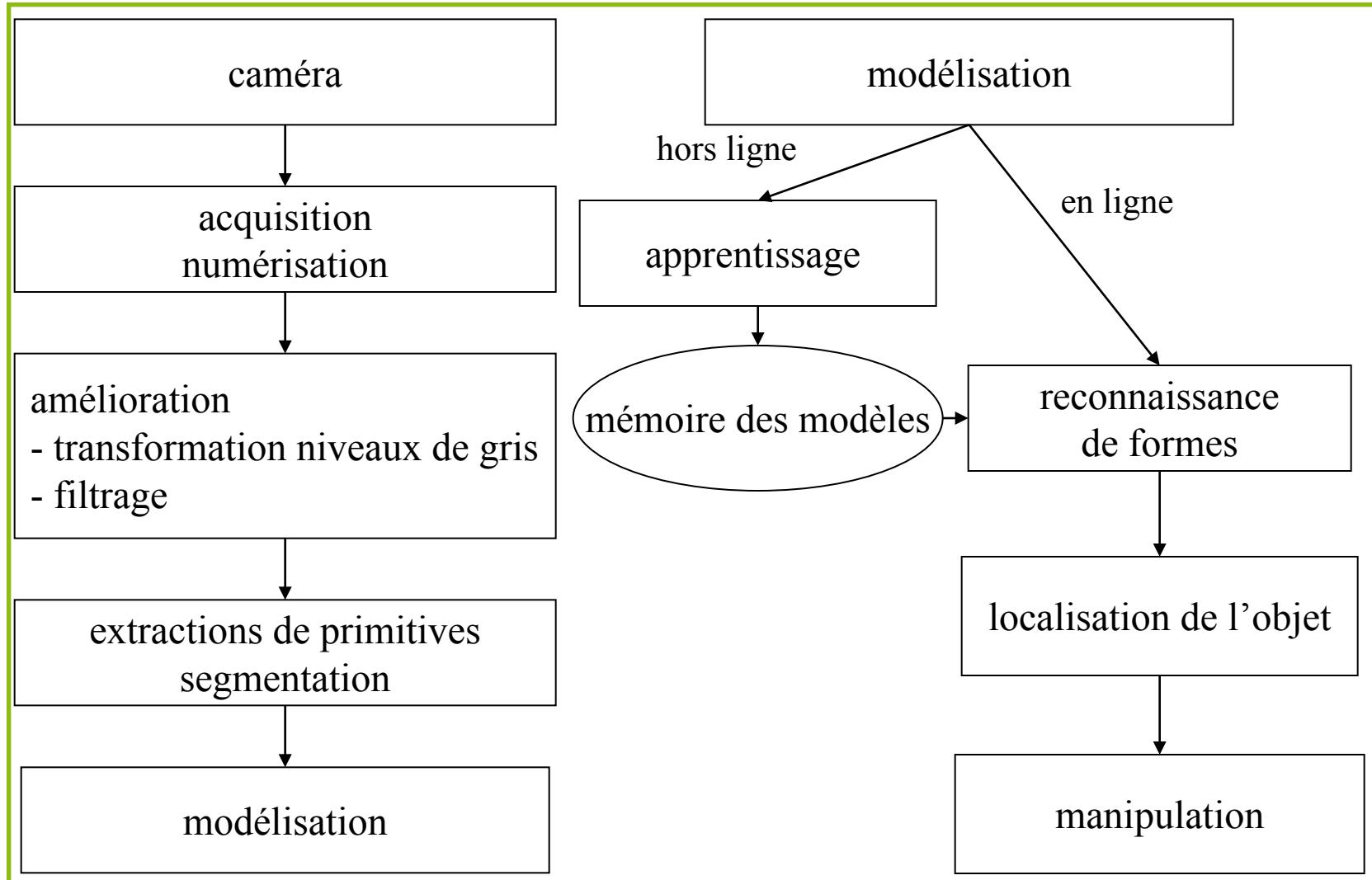
- Courbure d'un contour discret

$$\zeta(\vec{C}(t)) = \frac{\left\| \frac{\partial \vec{C}(t)}{\partial t} \wedge \frac{\partial^2 \vec{C}(t)}{\partial t^2} \right\|}{\left\| \frac{\partial \vec{C}(t)}{\partial t} \right\|^3} = \frac{\left(\frac{\partial x(t)}{\partial t} \cdot \frac{\partial^2 y(t)}{\partial t^2} \right) - \left(\frac{\partial^2 x(t)}{\partial t^2} \cdot \frac{\partial y(t)}{\partial t} \right)}{\left(\sqrt{\left(\frac{\partial x(t)}{\partial t} \right)^2 + \left(\frac{\partial y(t)}{\partial t} \right)^2} \right)^3} \text{ avec } \vec{C}(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}$$

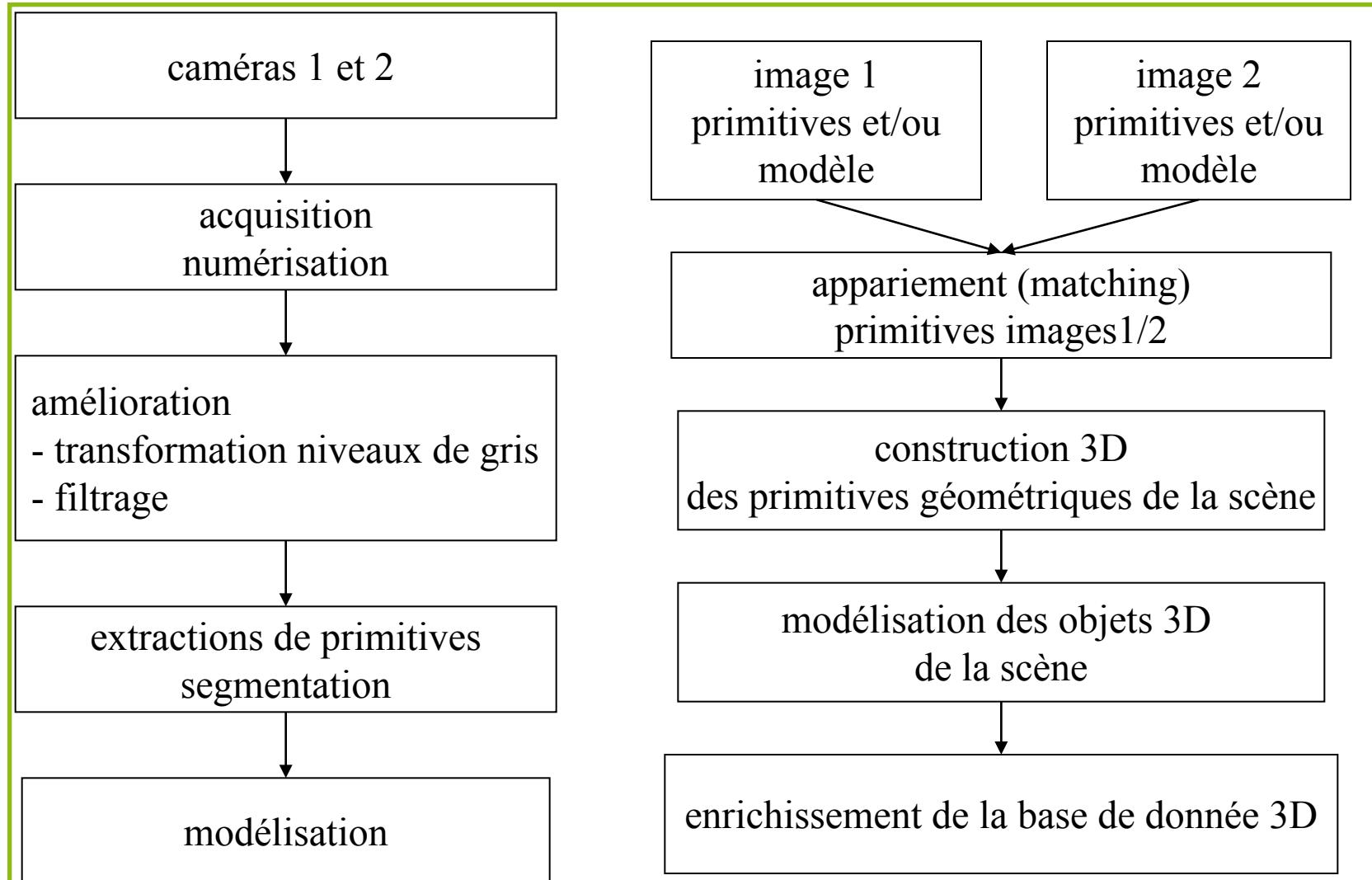
- Points de forte courbure \Rightarrow filtrage nécessaire



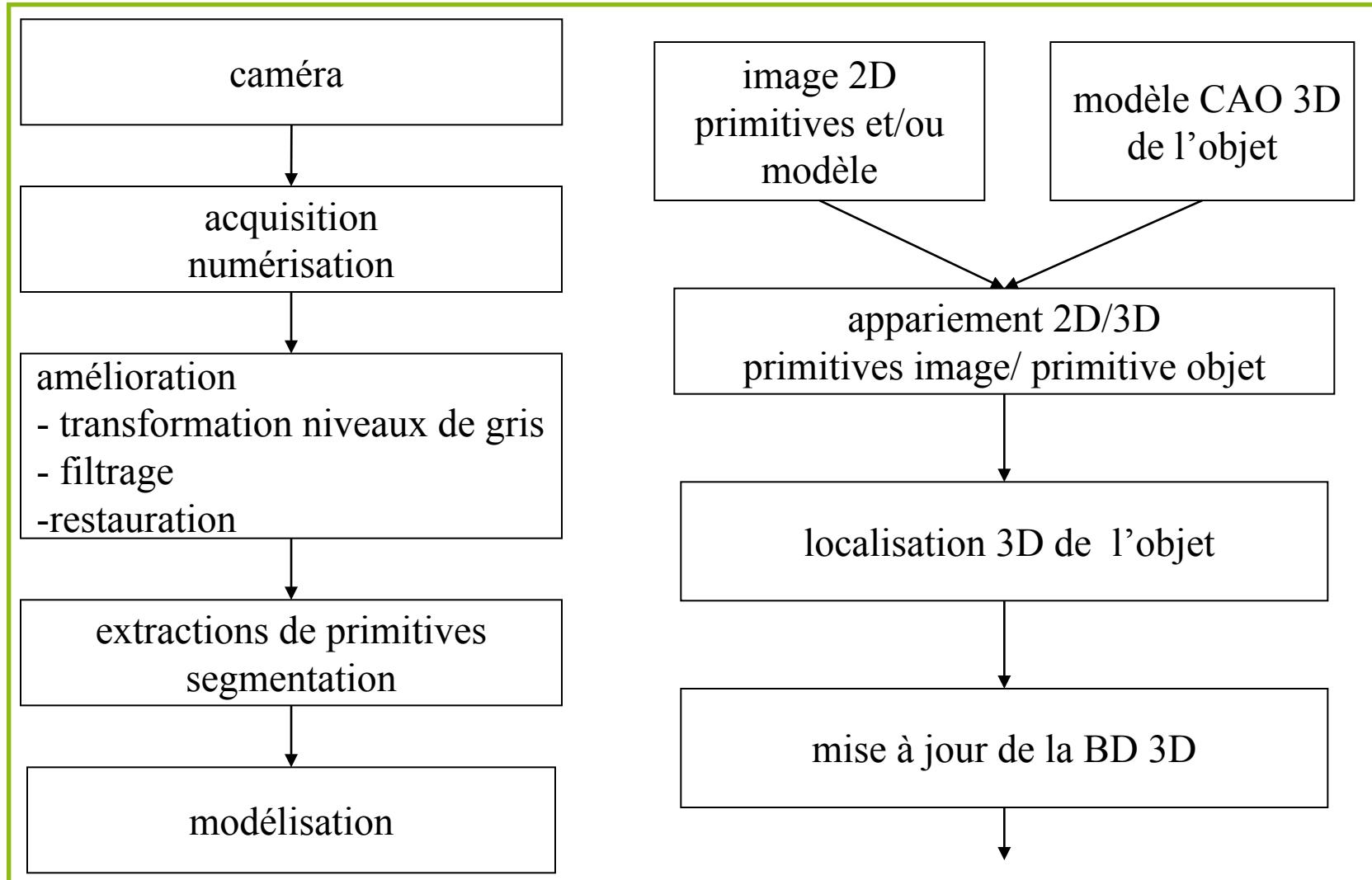
Chaîne de traitements pour application industrielle 2D



Chaîne de traitements pour application 3D stéréovision pour enrichissement de BD CAO 3D



Chaîne de traitements pour application 3D mise à jour de BD CAO d'un environnement





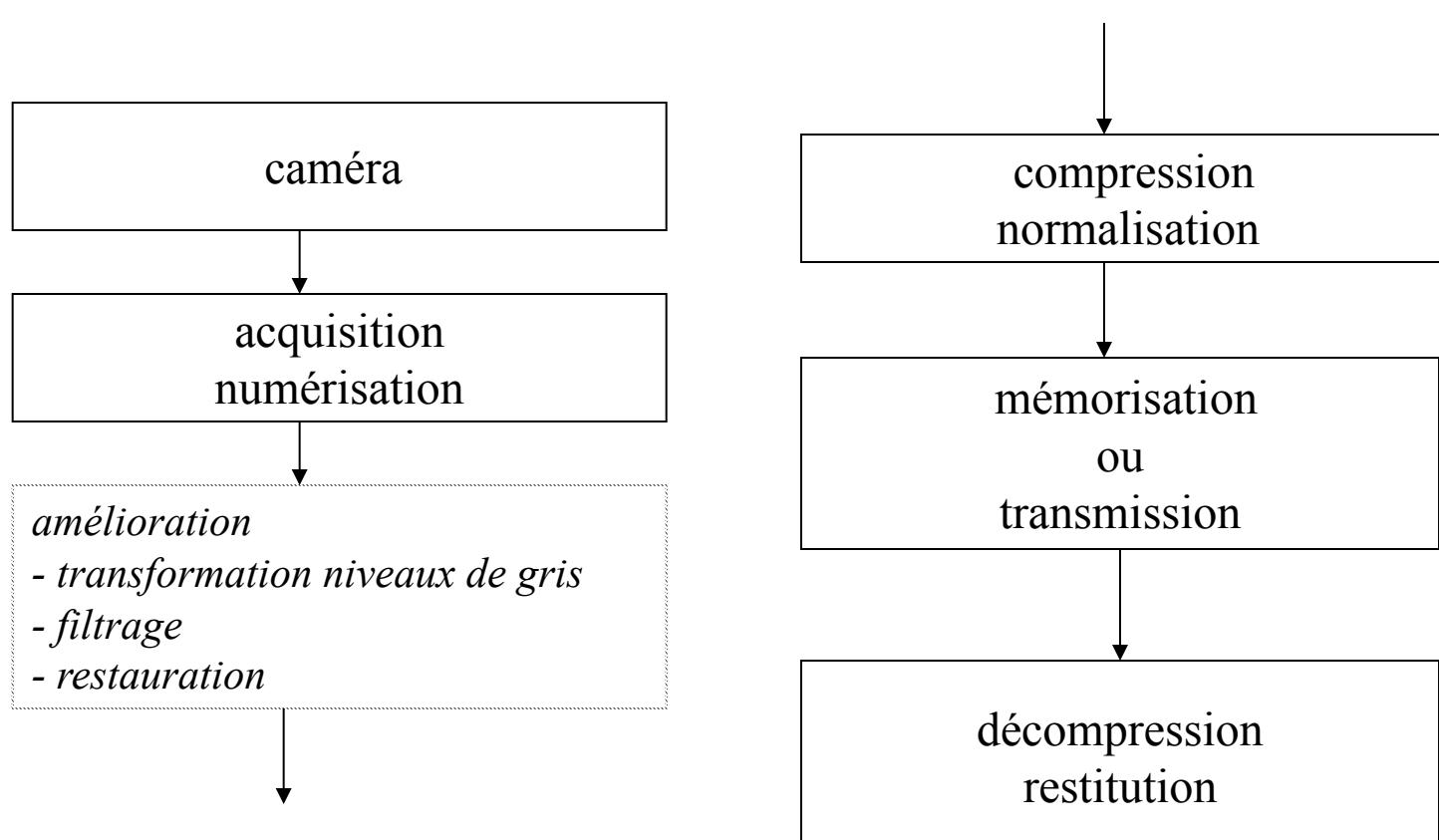
Donner un sens à une image
Donner un sens à une séquence
d'images

indexation d'images

Transmission/enregistrement d'images

- Diminuer le débit d'informations,
- Assurer une qualité d'images et de sons restitués compatibles avec les performances physiologiques de l'homme,
- assurer l'intégrité de la transmission ou de l'enregistrement

Chaîne de traitements pour transmission/mémorisation/restitution d'image



Transmission/enregistrement des images couleur codage source: luminance+chrominance

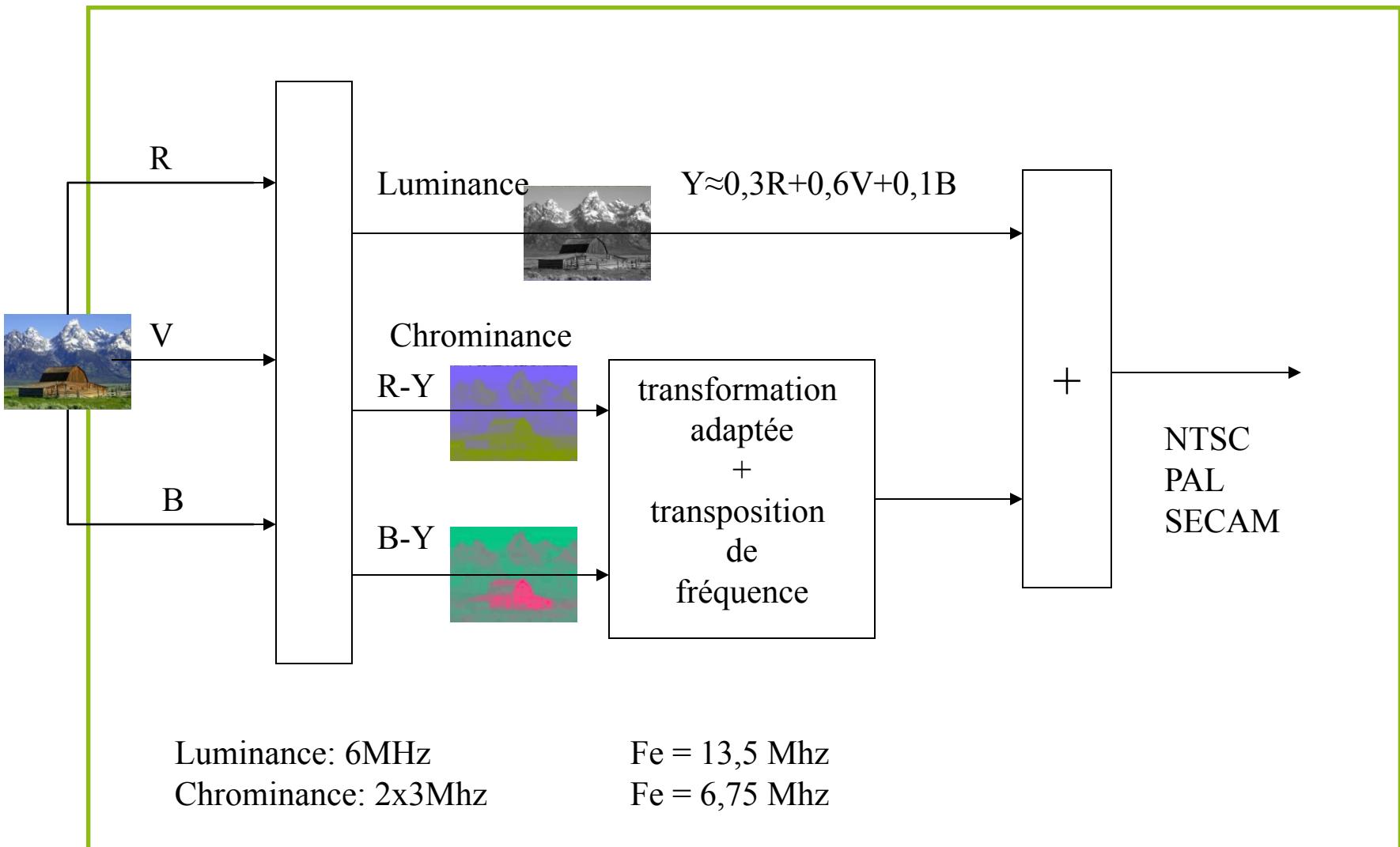
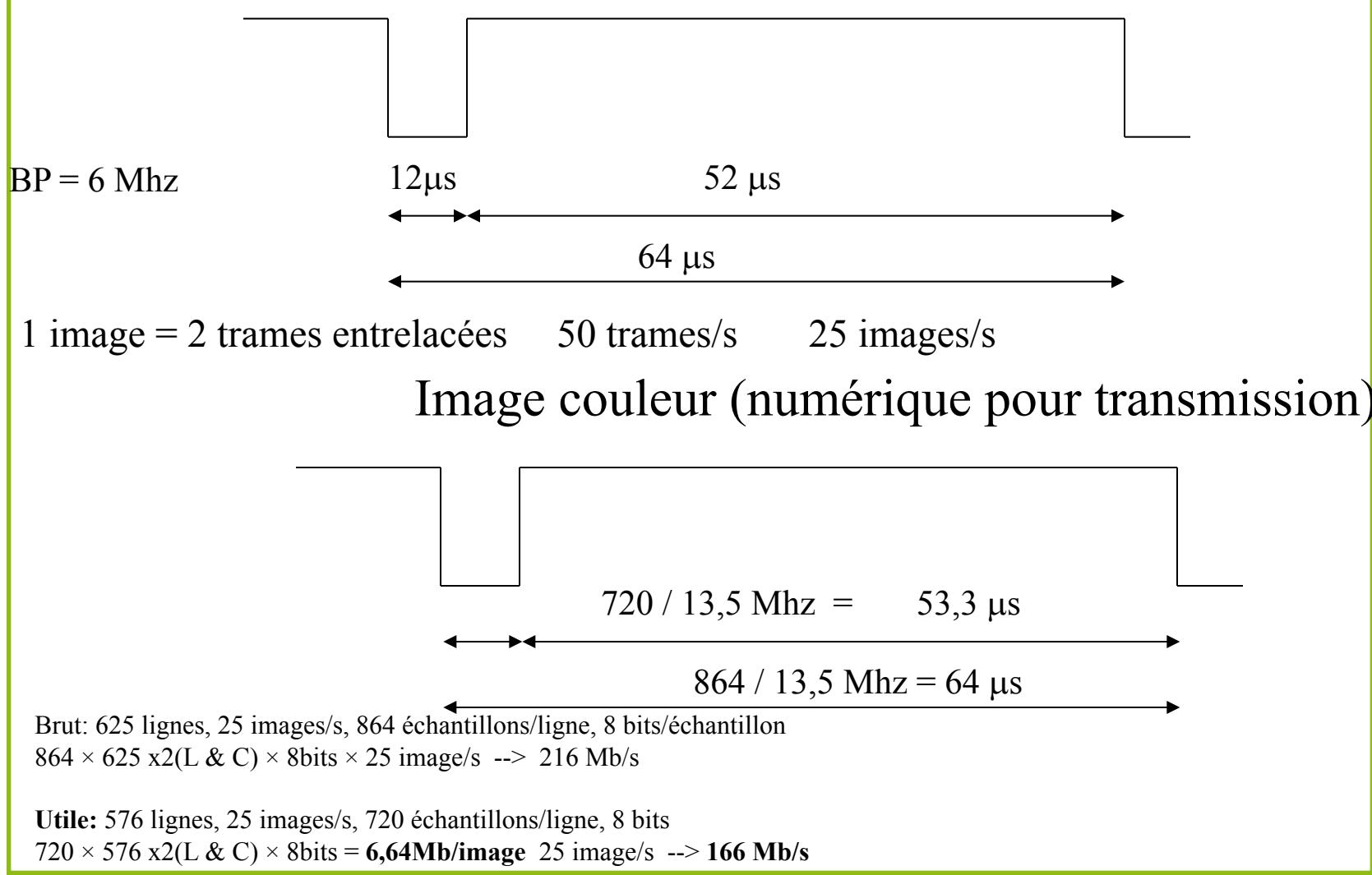
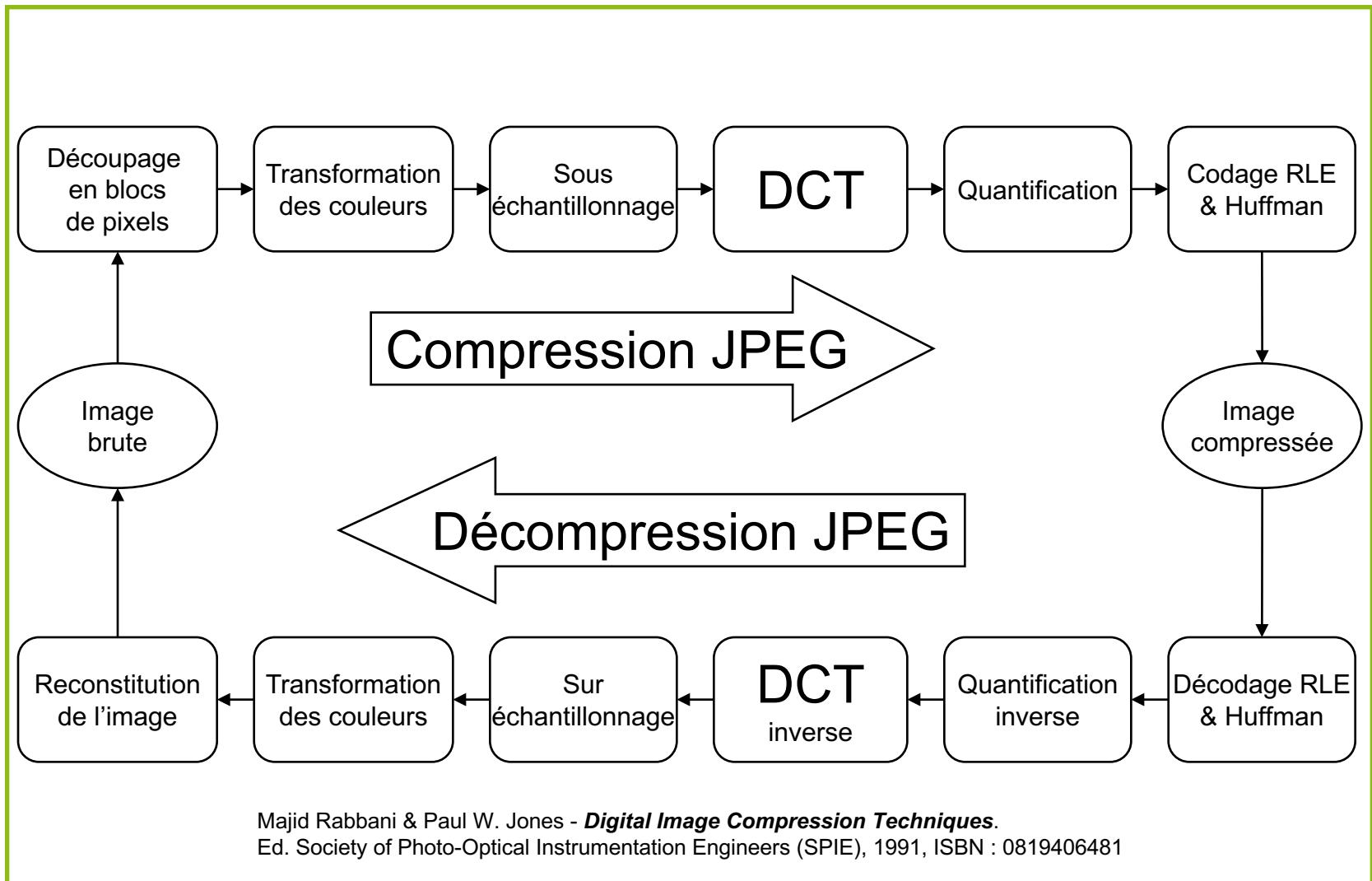


Image Noir et Blanc (analogique)



Compression JPEG : Joint Photographic Experts Group

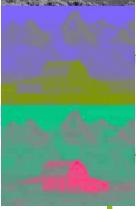


Compression JPEG

- Découpage en blocs

- 8×8 (64 pixels) ou 16×16 (256 pixels)

- Transformation des couleurs : RGB \rightarrow Luma/Chroma(s)



YC_bC_r

– YUV

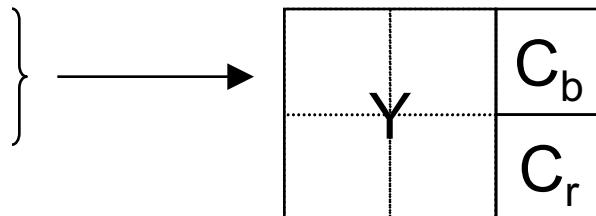
- Sous échantillonnage

- Exploite la plus faible sensibilité de l'œil à la chrominance
 - Sous échantillonnage des chrominances

- 2h1v = 2:1 en horizontal et 1:1 en vertical
 - 2h2v = 2:1 en horizontal et vertical
 - Macro-blocs

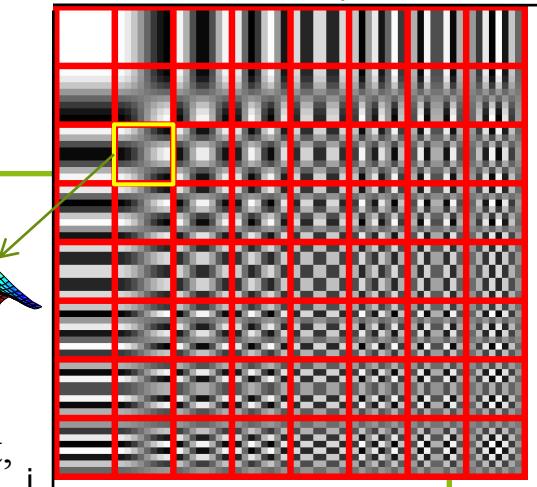
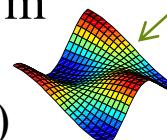
$$\begin{cases} Y = K_r R + (1 - K_r - K_b) G + K_b B \in [0 \dots 1] \\ P_b = \frac{1}{2} \left(\frac{B - Y}{1 - K_b} \right) \in [-0.5 \dots 0.5] \\ P_r = \frac{1}{2} \left(\frac{R - Y}{1 - K_r} \right) \in [-0.5 \dots 0.5] \end{cases}$$

$$K_r = 0.2999 \text{ et } K_b = 0.114 \Rightarrow (1 - K_r - K_b) = 0.587$$
$$R, G, B \in [0 \dots 1]$$



Compression JPEG

- Transformée DCT : Discrete Cosine Transform
 - Variante de la transformée de Fourier
 - Appliquée par bloc et par « canal » (Y, C_b, C_r)
 - Bloc DCT : taille $N \times N$
 - Carte de fréquences : indices = rapidité de changement, valeur = écart associé au changement de valeur



$$DCT(i, j) = \frac{2}{N} C(i) C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} I(x, y) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right)$$

$$C(k) = \begin{cases} \frac{1}{\sqrt{2}} & \text{pour } k = 0 \\ 1 & \text{pour } k > 0 \end{cases}$$

$$I = \begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

DCT

DCT⁻¹

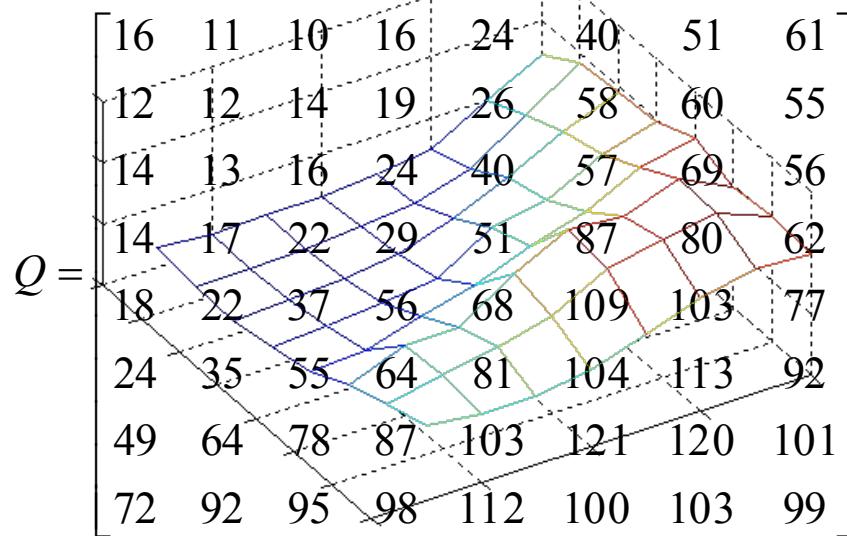
$$F = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

$$I(x, y) = \frac{2}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} C(i) C(j) DCT(i, j) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right)$$

Compression JPEG

- Quantification
 - Conserver le nombre minimum de coefficients non nuls dans F
 - Perte d'information
 - Application d'une matrice de quantification Q pour quantifier le bloc DCT :

$$F^*(i, j) = \frac{F(i, j) + \text{floor}(Q(i, j)/2)}{Q(i, j)} \approx \text{round}\left(\frac{F(i, j)}{Q(i, j)}\right)$$



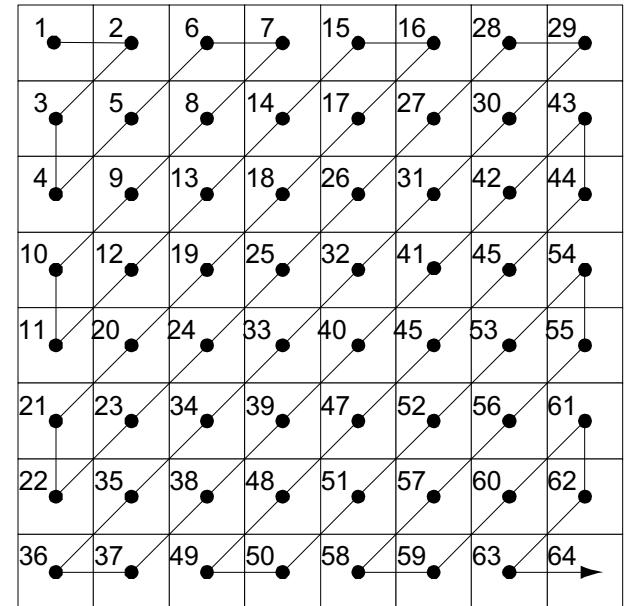
$$F^* = \begin{bmatrix} 79 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- « Déquantification » : $\hat{F}(i, j) = F^*(i, j)Q(i, j)$

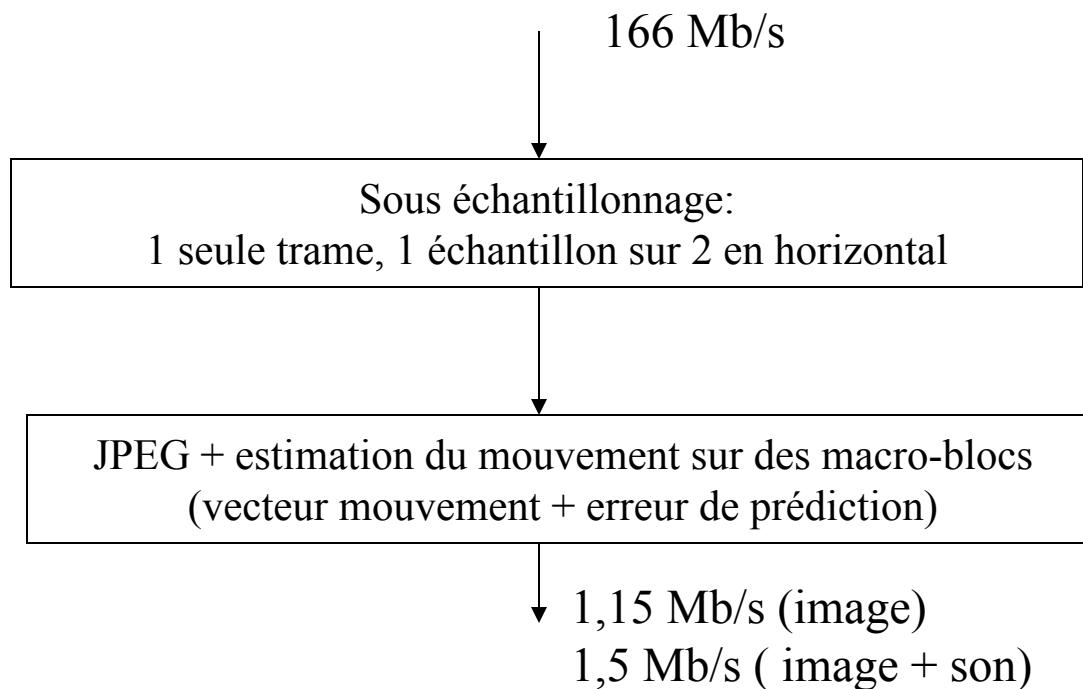
Compression JPEG

- Codage, compression RLE et Huffman
 - Parcours de F^* suivant les anti-diagonales pour obtenir une séquence commençant par les basses fréquences
 - Séquence
 - 79,0,-2,-1,-1,-1,0,0,-1,EOB
 - Compression Run Length (RLE) des zéros de la séquence
 - Codage entropique de type Huffman ou Codage Arithmétique
- Stockage
 - Tables de quantification
 - Dictionnaire des symboles du codage entropique
 - Blocs compressés

$$F^* = \begin{bmatrix} 79 & 0 & -1 & 0 \\ -2 & -1 & 0 & 0 \\ -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$



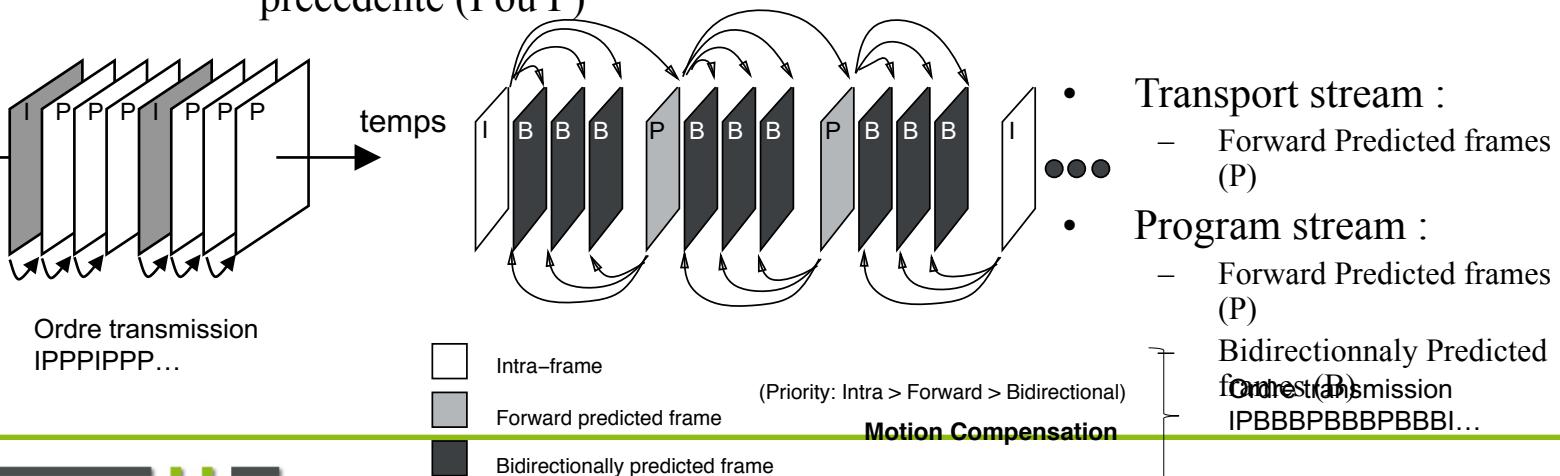
Compression MPEG (Motion Picture Expert Group)



MPEG audio couche 3 --> MP3

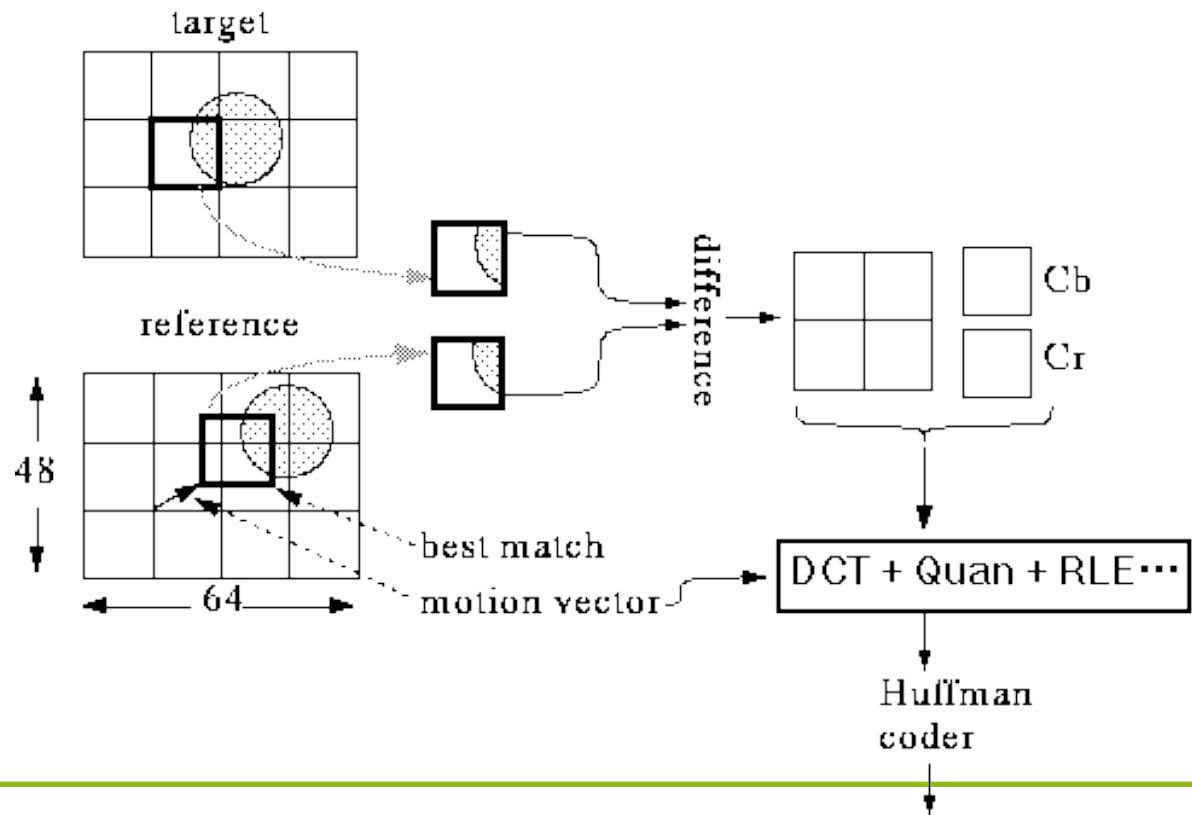
Compression MPEG (Motion Picture Expert Group)

- Compression vidéo
 - Compression intra-frame : réduction de la redondance spatiale
 - JPEG
 - Compression inter-frame : réduction de la redondance spatiale et temporelle
 - H261
 - MPEG(s)
 - Types de frames
 - I-frames : intra-frames encodées en JPEG
 - P-frames : inter-frames prédictes par pseudo-différence avec la frame précédente (I ou P)

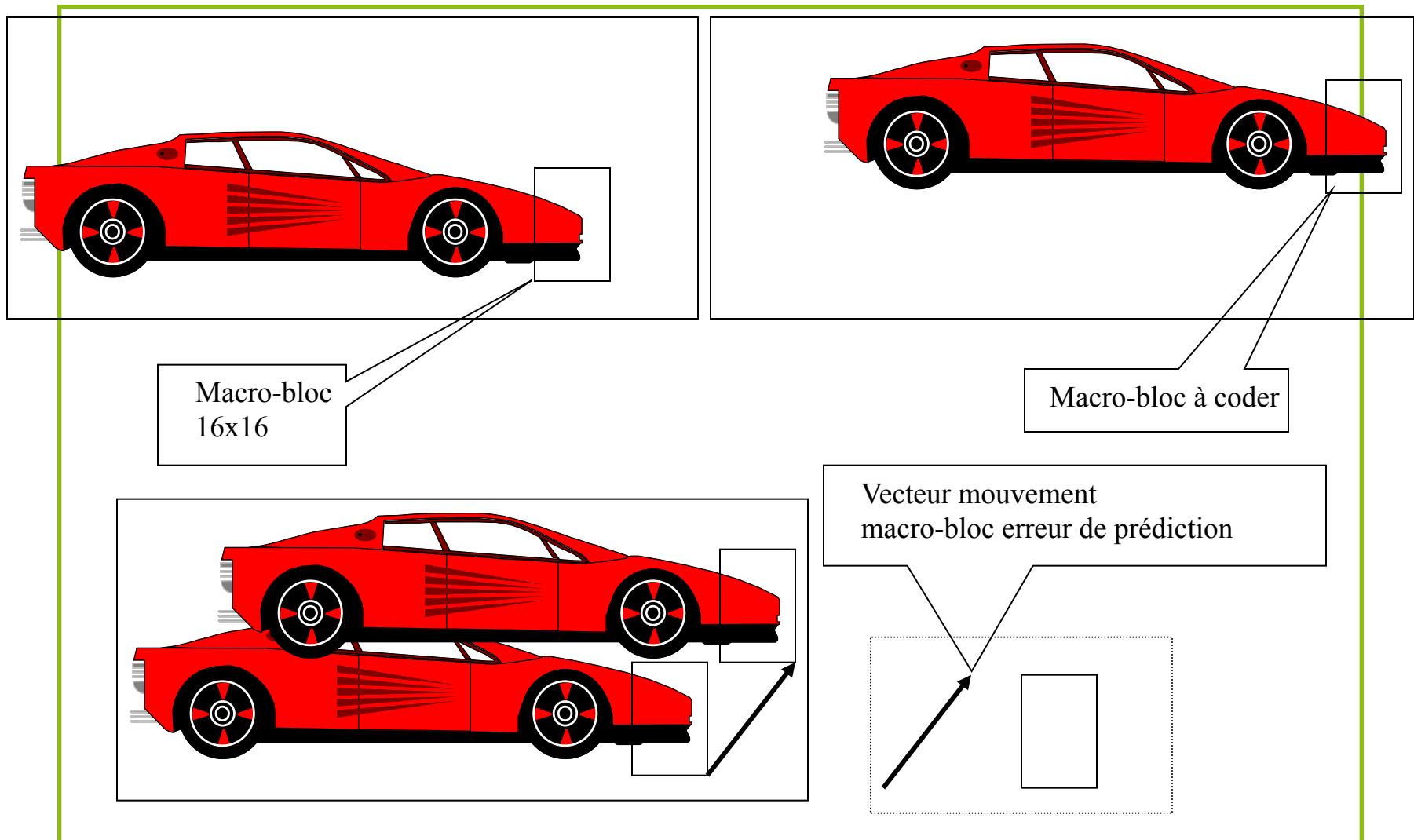


Compression MPEG : codage inter-frames

- Block-based motion compensated prediction
 - Reference = image précédente, Target = image à encoder
 - Encodage de la différence uniquement + motion vector

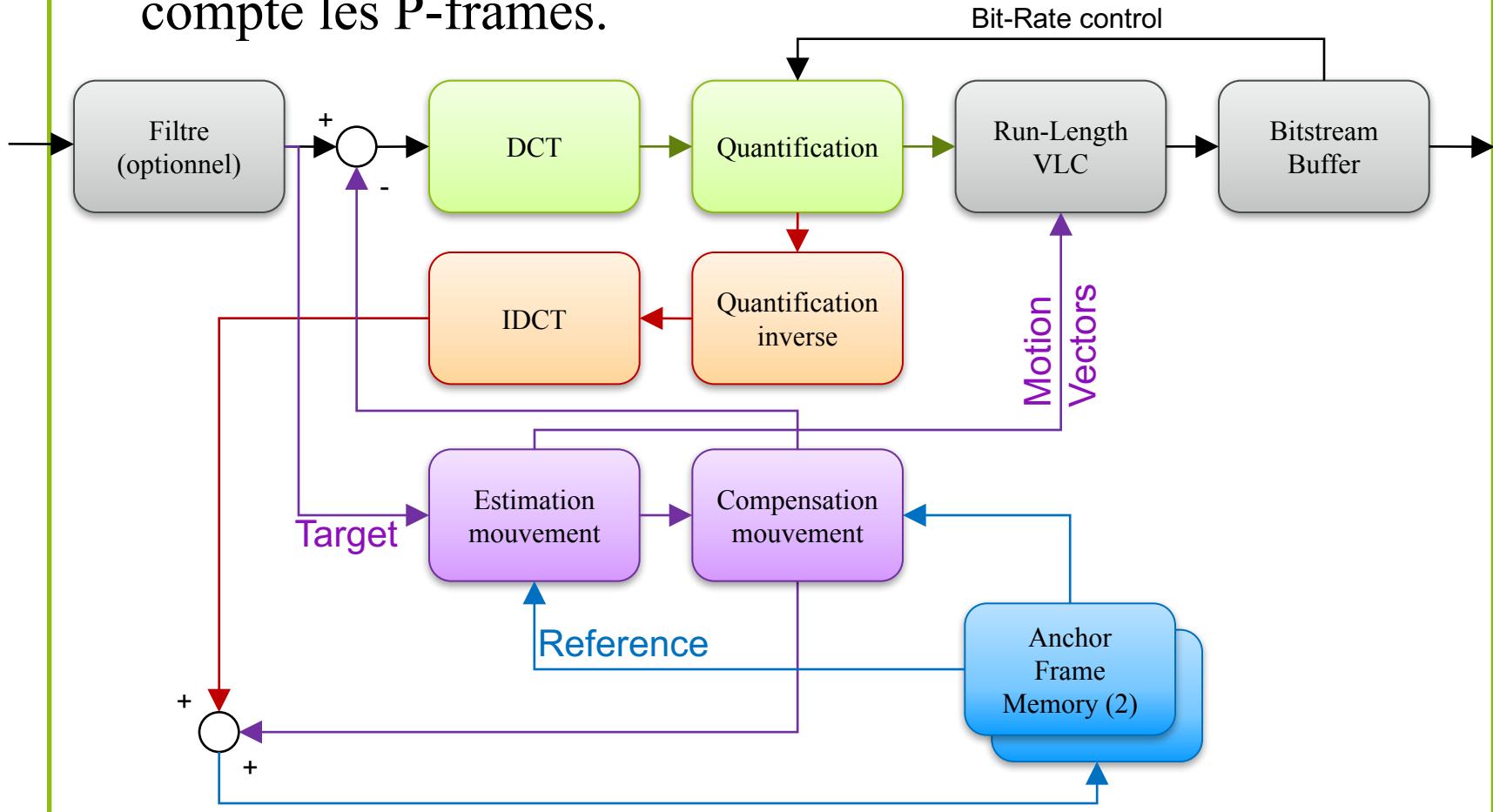


Prédiction mouvement Macro blocs MPEG

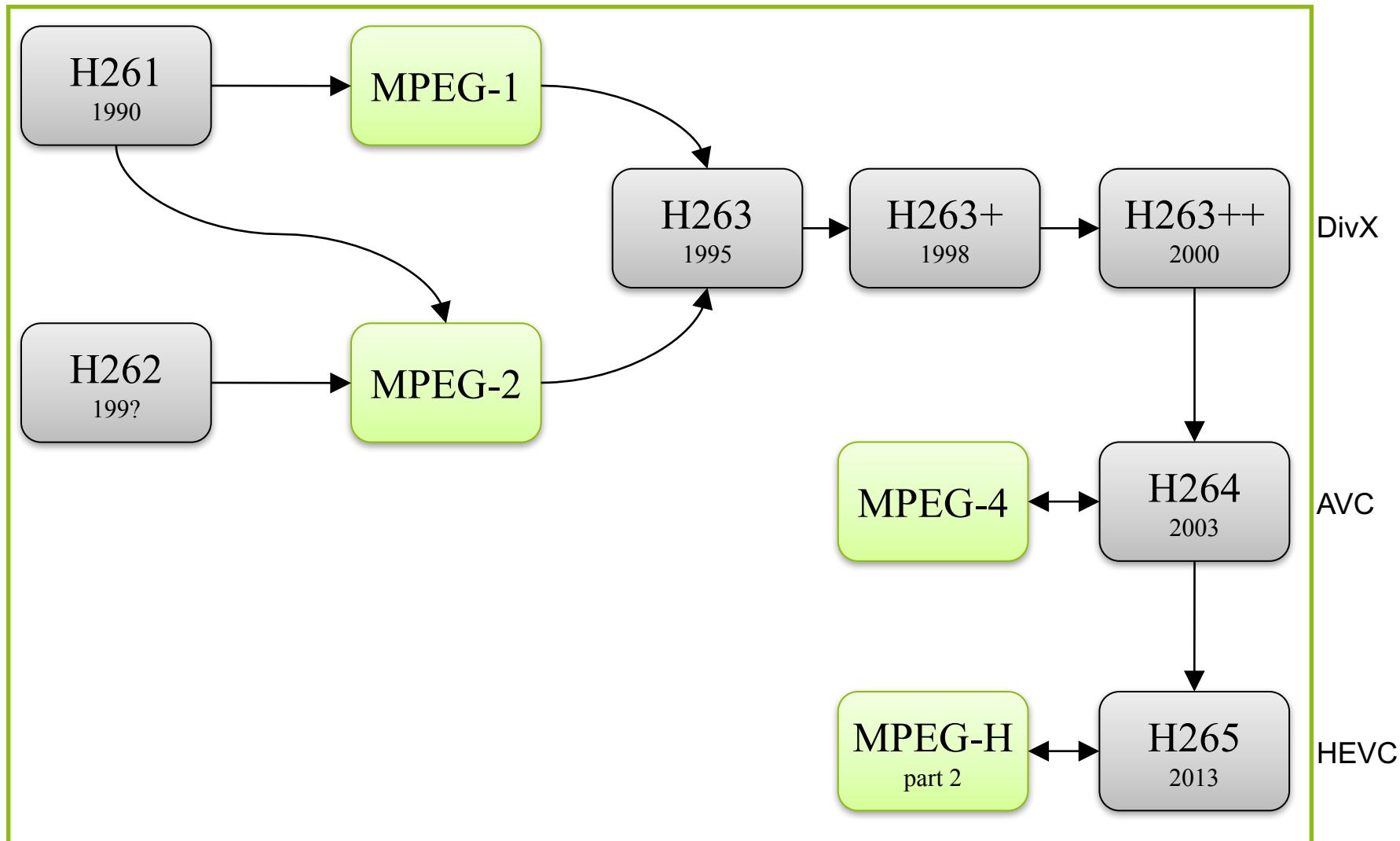


Compression MPEG : codage inter-frames

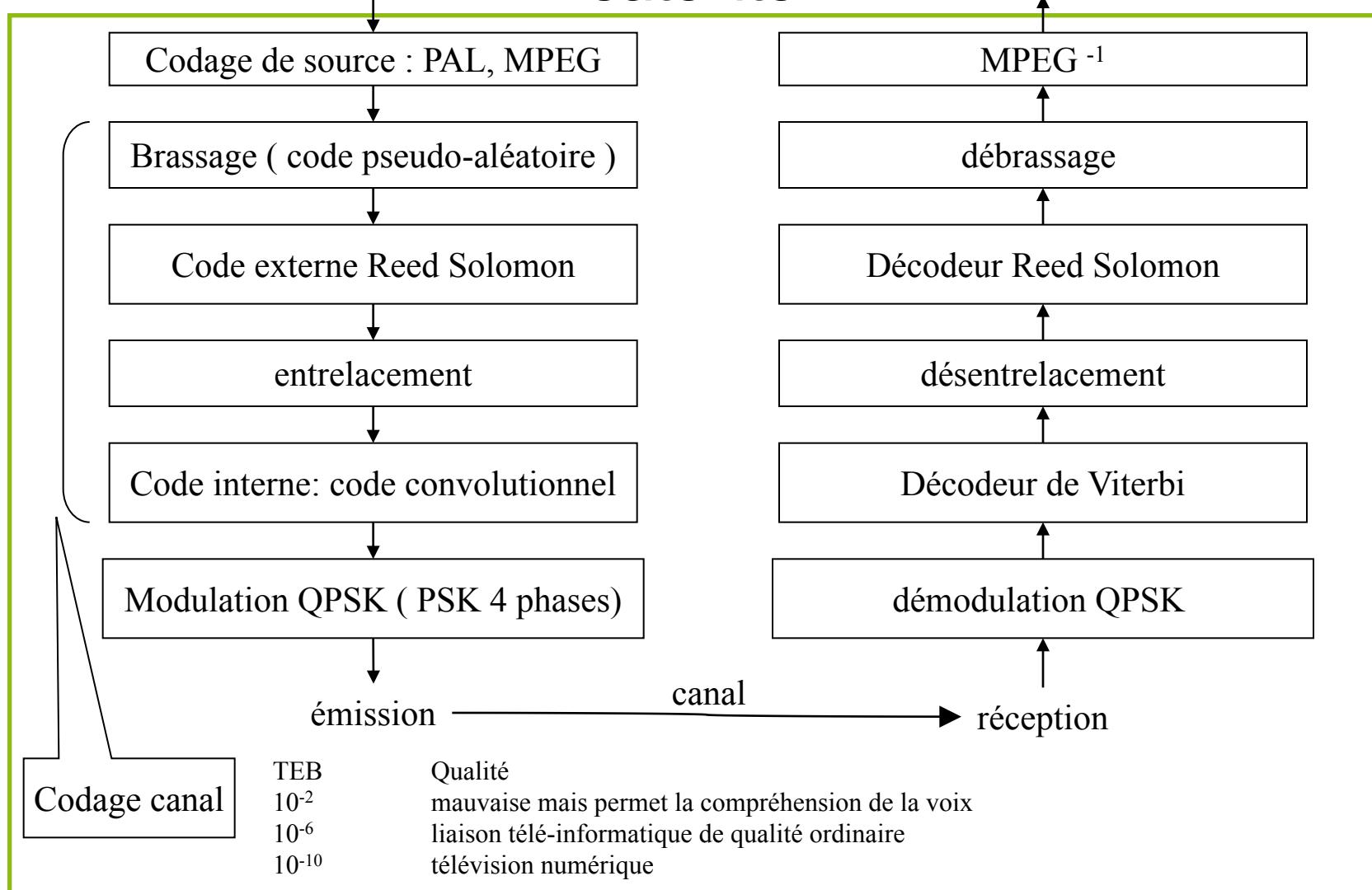
- Schéma de compression I-frames modifié pour prendre en compte les P-frames.



Compression MPEG : évolutions



Transmission de données (images et son) par satellite



Annexes

- Techniques de compression d'images
- Recherche de points d'intérêt dans une image
- Bibliographie

Techniques de compression d'images

- Compressions avec perte d'information
 - transformée Cosinus + seuillage --> JPEG
 - estimation du mouvement --> MPEG
 - quantification vectorielle
 - dictionnaires de blocs typiques
 - déterminer le bloc le + proche
 - transmettre l 'index du bloc
- Compressions sans perte d'information
 - entropique ou Huffman
 - codage de longueurs de saut (RLE) : nombre de 0, valeur (ou nbr 0, nbr 1)
 - LZF (Lempel-Ziv-Welch) (présent dans .gif, .tif, .pdf)
 - constitution d'un dictionnaire de groupes de bits
 - transmission de l'adresse du groupe de bits

Codage entropique (1)

Symboles équiprobables

nombre de bits par symbole: n

nombre de symboles ou états: $N = 2^n$

quantité d'information par symbole: $I = \log_2 N$ bits/symboles,

cas $N \neq 2^n$: $n = \text{partie entière supérieure}(I)$

Alphabet de N symboles non équiprobables

le symbole $N^{\circ} i$ a probabilité d'occurrence p_i

la quantité d'information contenue dans le i ème symbole = $\log_2 \frac{1}{p_i}$

quantité moyenne d'information par symbole: $I = \sum_{i=1}^N p_i \log_2 \frac{1}{p_i} = - \sum_{i=1}^N p_i \log_2 p_i$

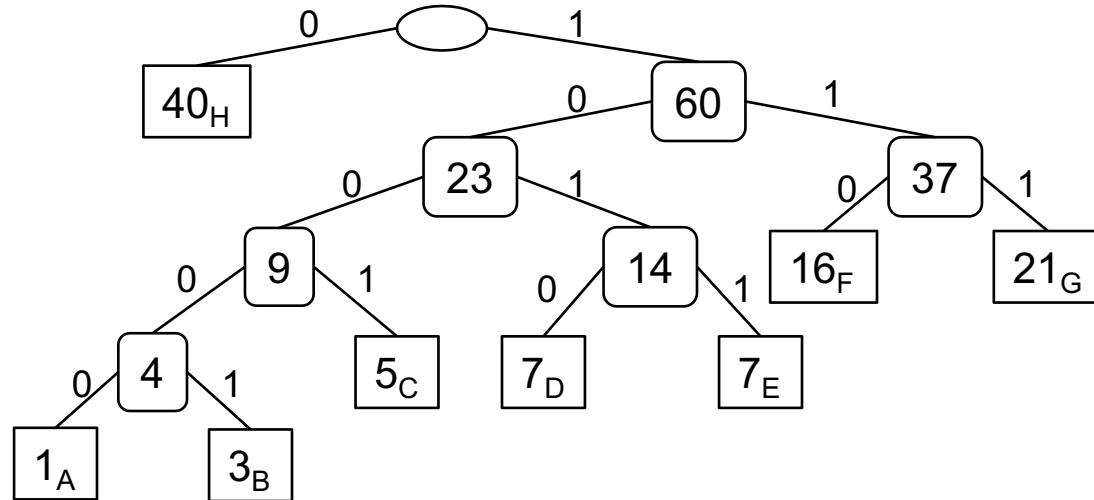
cas particulier: N symboles équiprobables: $p_i = \frac{1}{N}$ et $I = \log_2 N$

cas général: $\log_2 N \geq - \sum_{i=1}^N p_i \log_2 p_i$

Codage entropique ou de Huffman (2)

1. Déterminer les fréquences f_i associées à chaque symbole $p_i = f_i / \sum f_i$
2. classer les symboles dans l'ordre décroissant des f_i
3. construire un arbre des feuilles vers la racine en groupant successivement les symboles de plus faible f_i et en faisant des regroupements de symboles selon le principe de séparation par dichotomie (fréquences égales ou voisines)
4. attribuer à chaque branche de l'arbre une valeur binaire de telle manière que plus la fréquence f_i est grande, plus le nombre de bits représentant le symbole est faible.

Symboles	A	B	C	D	E	F	G	H
Fréquences % (p_i)	1	3	5	7	7	16	21	40



173

Vision

Bibliographie générale

Analyse d'images : filtrage et segmentation.
sous la direction de J.P. Coquerez, S. Philipp
Masson 1995

Les systèmes de vision
sous la direction de Jean Michel Jolion
Hermes, traité IC2 – Série Traitement du
signal et de l'image. 2001

Signaux et images sous Matlab
méthodes, applications et exercices corrigés
sous la direction de Gérard Blanchet et
Maurice Charbit
Hermes 2001

Introduction au traitement d'images
Gilles Burel
Hermes 2001

Vision par ordinateur. Outils fondamentaux.
R. Horaud, O. Monga
Hermes 1993

Three-dimensional computer vision. A
geometric Viewpoint .
O. Faugeras
MIT Press 1993

Perception de l'environnement en robotique
F. Chavand, E. Colle
Hermes 1998

Digital Image Processing
R.C. GONZALEZ et R.E. WOODS
Addison-Wesley Editors (1992)

Traitement Numérique des Images
M. KUNT, G. GRANLUND et M.KOCHER
Presses Polytechniques et Universitaires
Romandes (1993)

Digital Picture Processing
A. ROSENFELD et A.C.KAK
Vol 1, Academic Press, Londres (1982)



Vision 3D

Ecole Nationale Supérieure d'Informatique
pour l'Industrie et l'Entreprise

Florent Chavand – 1998 - 2007

David Roussel – 2010

Capteurs et procédés de mesure

- Mesures absolues
 - *Informations de profondeur* \Rightarrow position/orientation de l'objet
 - Stéréocopie & photogrammétrie
 - Temps de vol
 - Shape from
 - Focus / Defocus
 - Structured Light : Projection de lumière structurée & triangulation laser
 - Informations volumiques
 - Tomographie
- Mesures relatives
 - *Géométrie [locale] de la surface de l'objet mais pas sa position*
 - Shape from ...
 - Shading
 - Textures

Approches de la reconstruction 3D

- Mise en correspondance
(top - down)

- Recherche des occurrences d 'un type d 'objet connu, et replacement dans l 'espace de la scène.
- Primitives de haut niveau : objets prédéfinis
- Exemple :

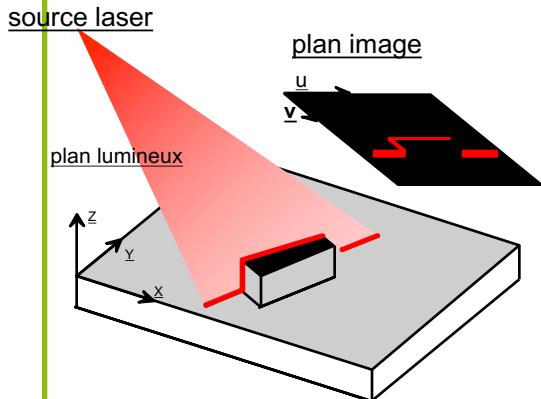
- cylindres [Viala 92]
- cylindres généralisés [Ponce 89]

- Constructives
(bottom - up)

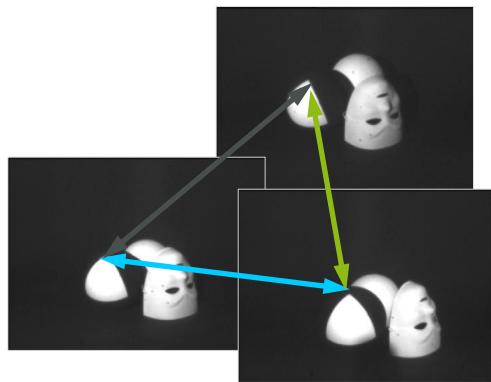
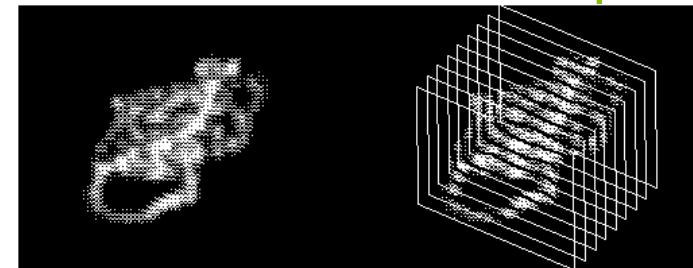
- Reconstruction sans utiliser d 'objets connus a priori
- Utilisation de primitives de reconstruction de bas niveau, dont l 'assemblage construit une entité de plus haut niveau.

Formes libres ⇒ Approche constructive

La vision pour la reconstruction 3D



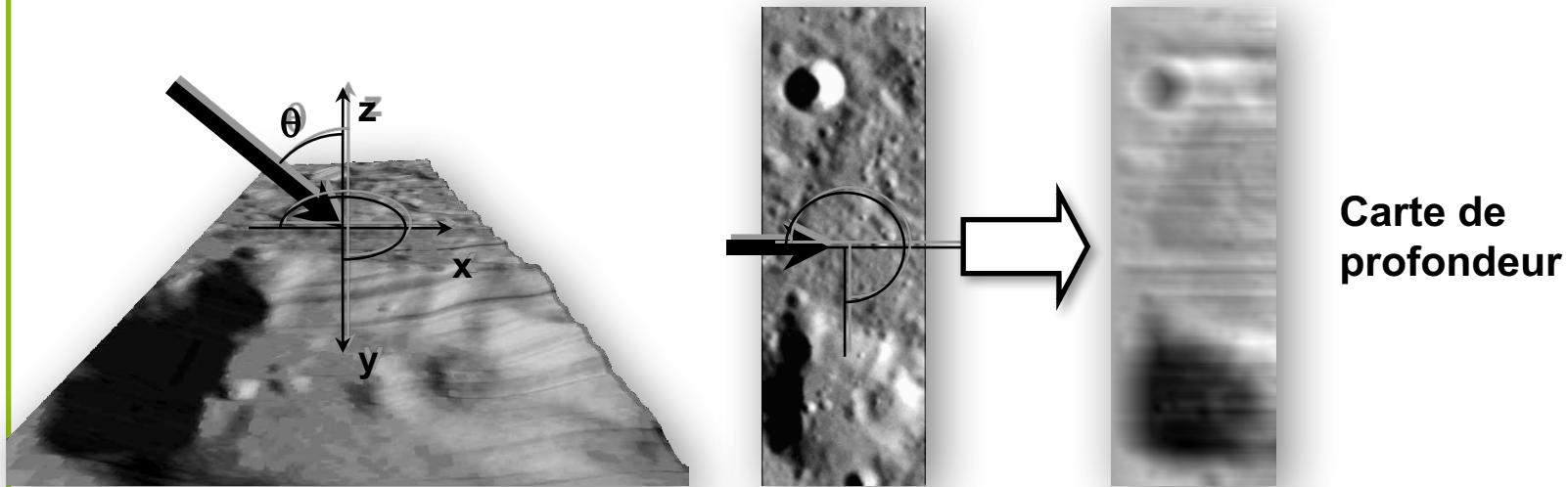
- Vision Active
 - Image + source
 - Laser
 - Tomographie
 - Shape from shading



- Vision Passive
 - Exploitation des différences entre les images
 - Vision multi-oculaire (stéréo-vision)
 - Vision monoculaire en mouvement

Shape from Shading (1/2)

- Principe : Approche photoclinométrique
 - a : Albédo constant et connu
 - Source(s) lumineuse(s) connue(s) $I(x, y) = aR[p(x, y), q(x, y)]$
$$p(x, y) = \frac{\partial z(x, y)}{\partial x}, q(x, y) = \frac{\partial z(x, y)}{\partial y}$$



[Horn, 1989] Horn, B. K. P. (1989). Shape from shading, chapter Obtaining shape from shading information, pages 123–171. Number 5. MIT Press, Cambridge, MA, USA.

Shape from Shading (2/2)

- Modèle d'éclairage diffus (Lambertien)

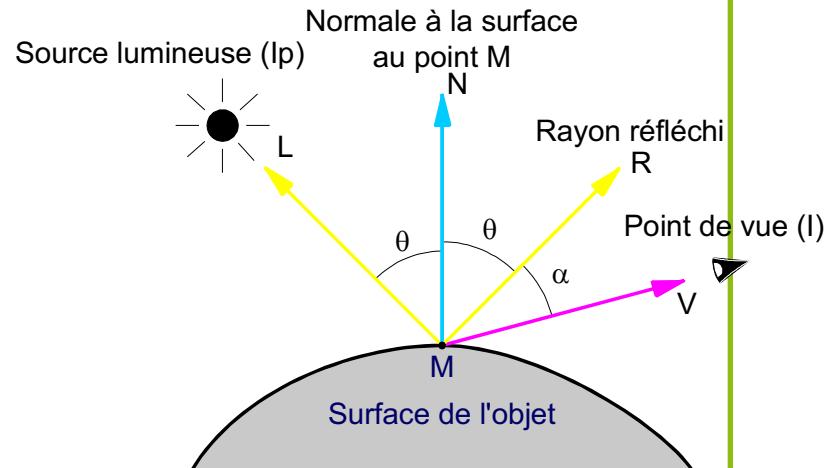
$$I(x, y) = I_p k_d \cos(\theta) \Rightarrow \theta = \alpha \cos\left(\frac{I(x, y)}{I_p k_d}\right)$$

$I(x, y)$ = Intensité lumineuse

I_p = Intensité lumière incidente

k_d = Coef de réflexion diffuse

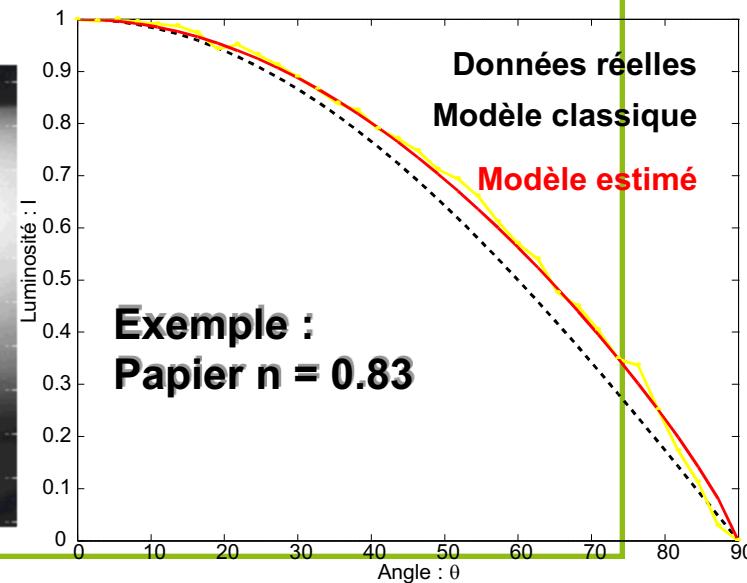
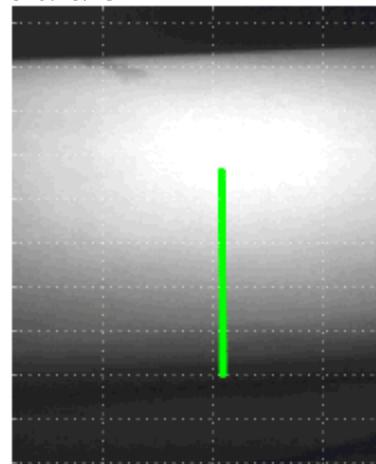
I_a = Intensité lumière ambiente



- Modèle d'éclairage pseudo diffus

$$I(x, y) = I_a + I_p k_d \cos^n(\theta)$$

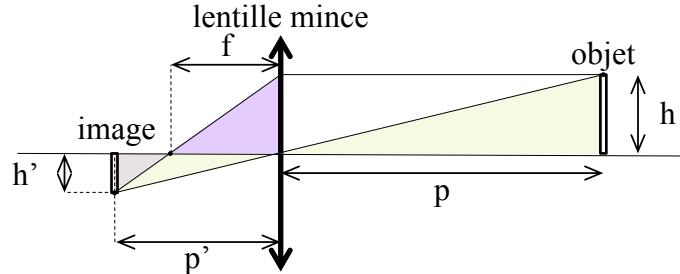
$$\Rightarrow \theta = \alpha \cos\left(\left(\frac{I(x, y) - I_a}{I_p k_d}\right)^{\frac{1}{n}}\right)$$



Exemple :
Papier n = 0.83

Shape / Depth from focus / defocus

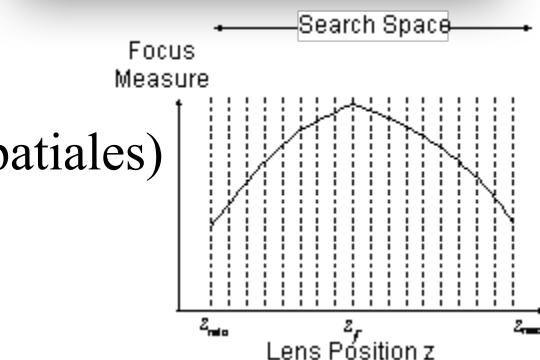
- Problématique : Obtenir une estimation 3D de la scène à partir de 2 ou plusieurs images obtenues en modifiant la focale ou la position du capteur
- Depth from focus (actif)



$$\frac{h'}{h} = \frac{p'}{p} = \frac{p'-f}{f} = \frac{p'}{f} - 1 \Leftrightarrow \frac{1}{p'} - \frac{1}{p} = \frac{1}{f} \Leftrightarrow \frac{1}{f} = \frac{1}{p} + \frac{1}{p'}$$

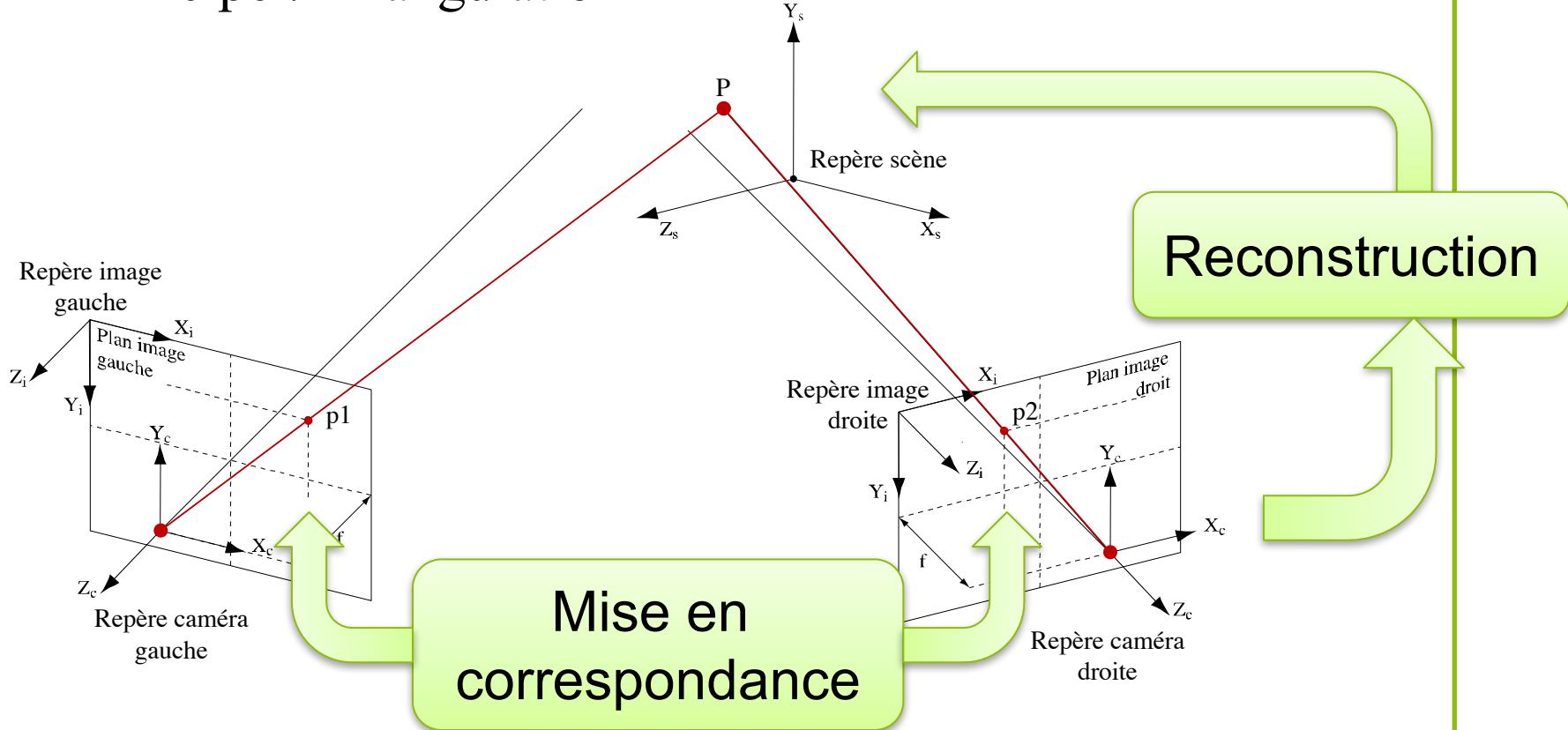
- Relations $p' \leftrightarrow p$
- Mesure de focus (\approx hautes freq. spatiales)

- Depth from defocus (passif)
 - Relations $p' \leftrightarrow p$
 - Mesure de défocus



Reconstruction stéréo

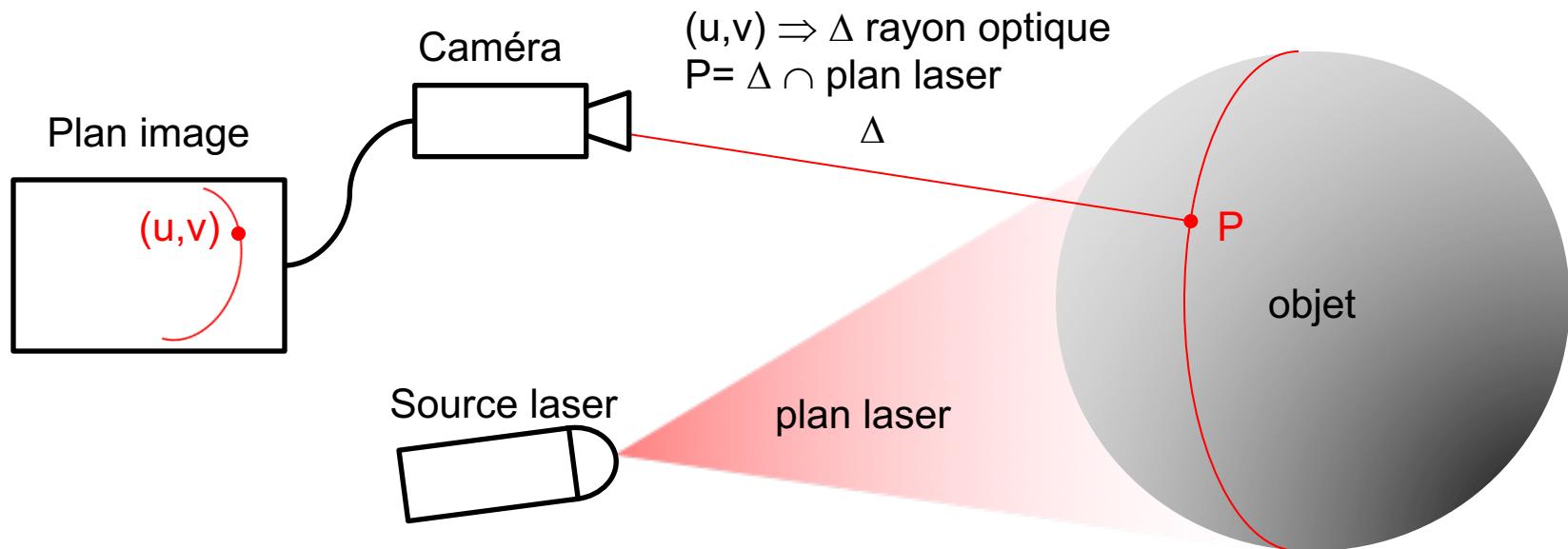
- Principe : Triangulation



Nécessite une calibration (ou une auto-calibration)

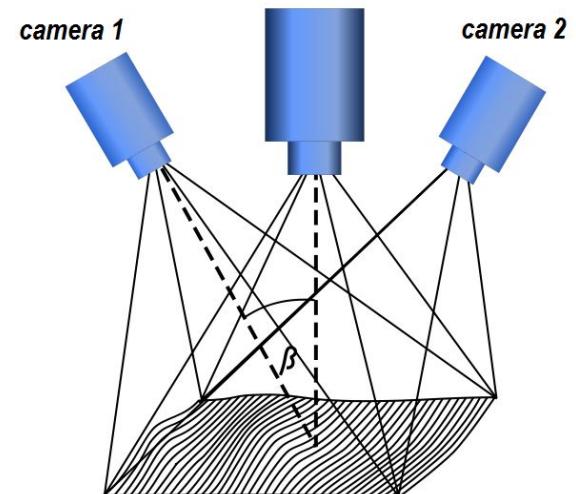
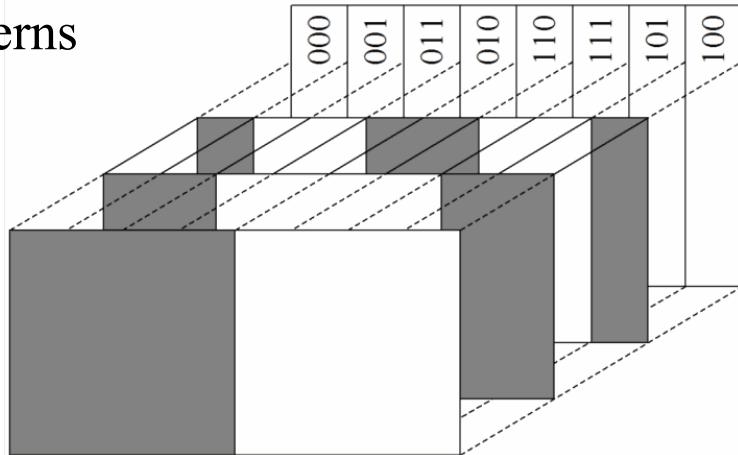
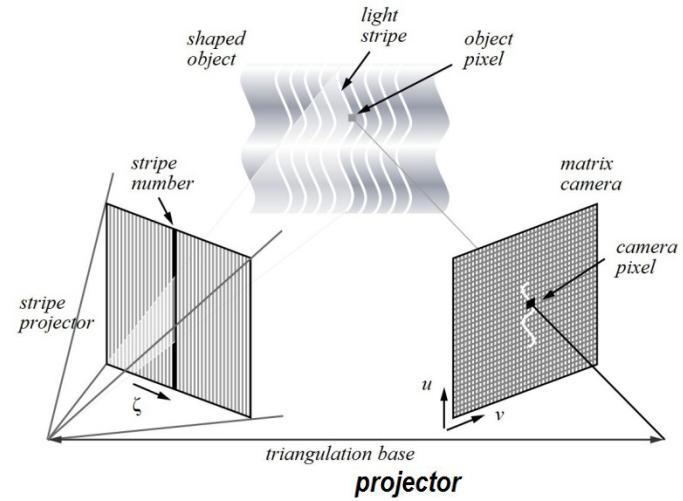
Shape from structured light (1/2)

- Projection Laser : rayon → point, plan → ligne, grille
- Exemple plan laser
 - Plan laser :
 - lentille cylindrique ou miroir tournant
 - Équation connue
 - Caméra calibrée



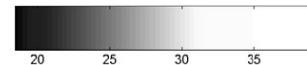
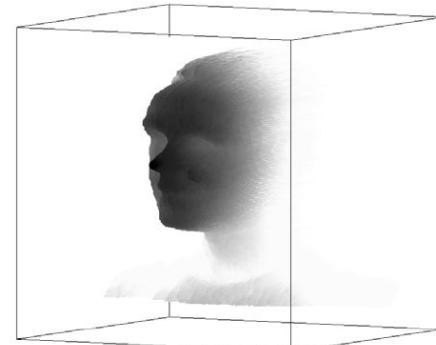
Shape from structured Light (2/2)

- Projection d'un ou plusieurs « patterns »
- Patterns
 - Interférence (lumière cohérente)
 - Projection
- Codage des patterns
 - 1 seul bit change entre deux patterns



Range Imaging : Time of Flight

- Principe : LIDAR
 - Émission d'une impulsion de lumière à t_0 de durée d
 - Mesure sur chaque pixel(i,j) de la lumière réfléchie de l'impulsion $t_0 + \Delta t(i,j)$: Δt indique alors la distance
- Technologies
 - TOF : impulsions mesure de Δt du signal réfléchi
 - Phase : Modulation continue à la fréquence f , mesure de déphasage du signal réfléchi
- Exemple
 - Objet à $D=2.5$ m $\Delta t=2D/c = 16.6$ ns
 - Range max pour $d = 50\text{ns}$: $cd/2 = 7.5$ m
 - Phase $\Phi = 4\pi Df/c$

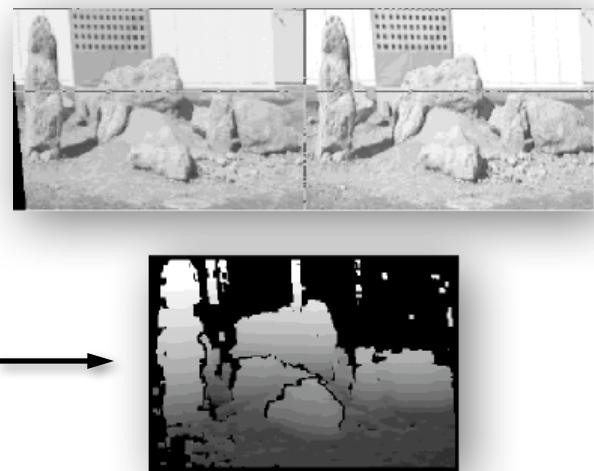


Méthodes de reconstruction

- Méthodes Globales :
traitent l'ensemble de(s) l'image(s).

- Shape from XXX
- Stéréo
 - Corrélation
 - Elastic Matching

} **Carte de profondeur**

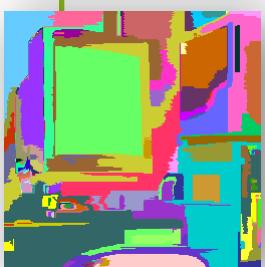
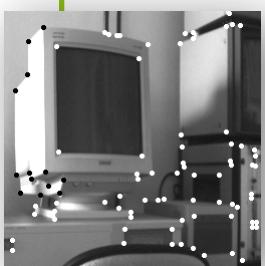


- Méthodes basées sur des primitives :
traitent uniquement les primitives extraites des images.

- Points particuliers
- Contours
- Régions

} **Conditionnent la reconstruction**

Primitives 2D / 3D

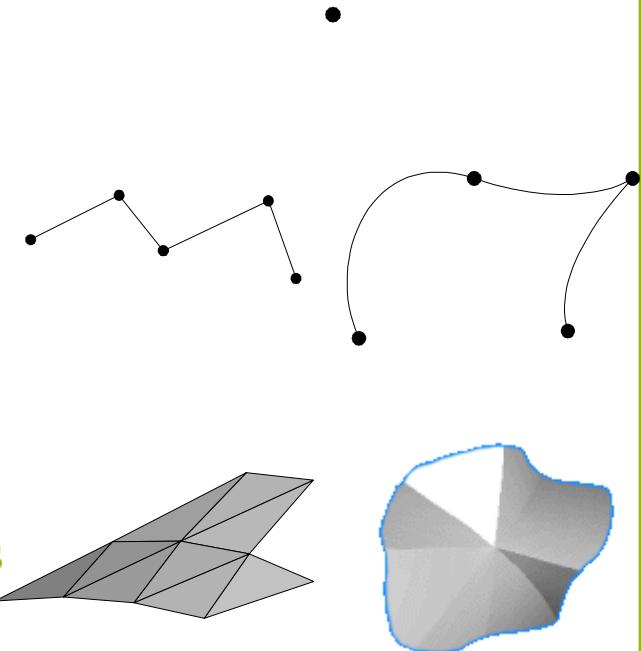


- Points d'intérêts
 - Sommet
 - ☹ Faibles critères de ressemblance¹
 - ☺ Disparité ponctuelle
- Les contours
 - Arête (segment, courbe)
 - ☺ Forts critères morphologiques
 - ☺ Disparité le long du contour
- Les régions
 - Face (facette, carreau)
 - ☺ Graphe d'adjacence des régions
 - ☹ Disparité ponctuelle grossière

Métrique 3D

Polyédrique

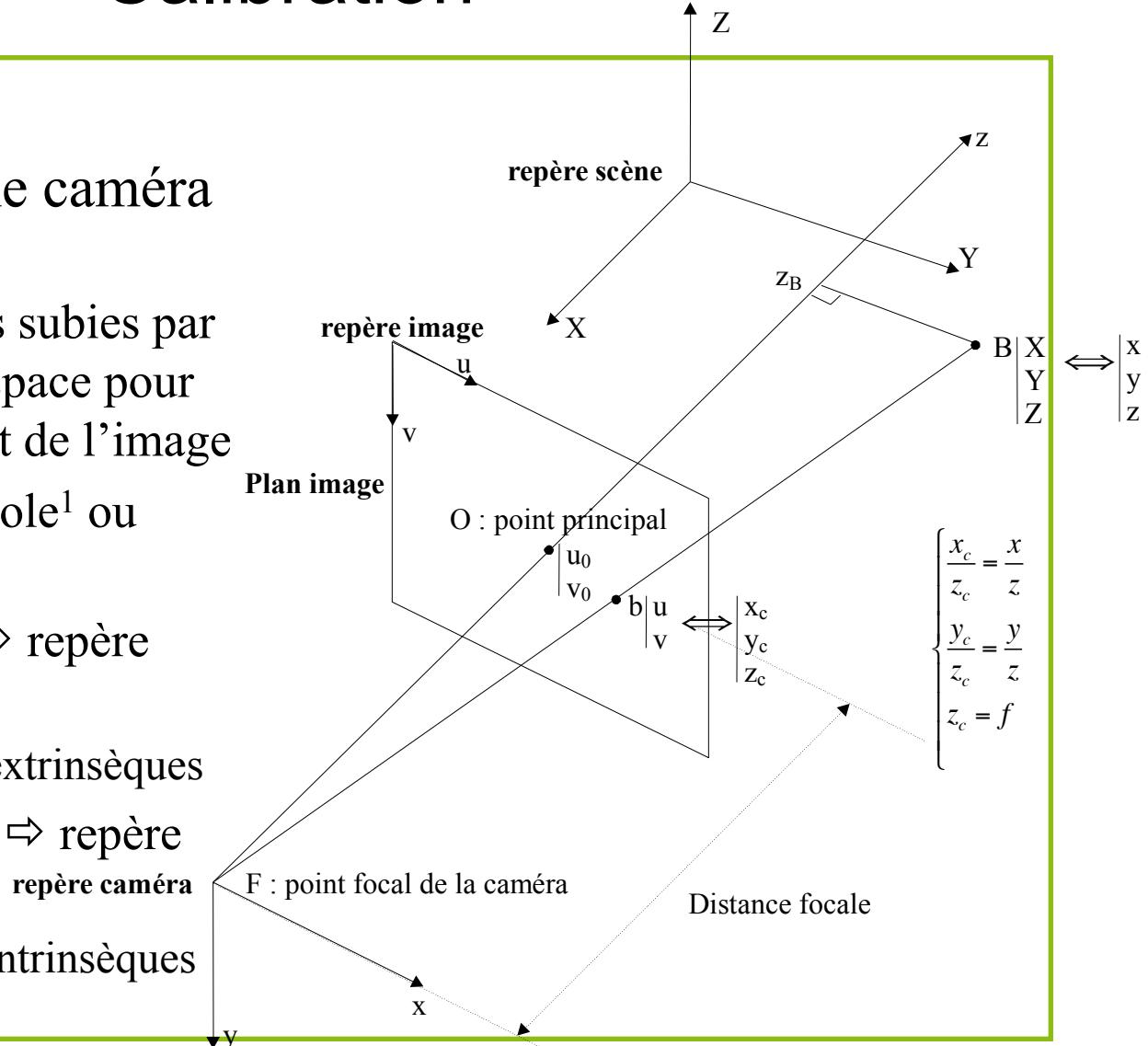
Formes libres



¹ en eux-mêmes, on peut toutefois utiliser leur voisinage

Calibration

- Estimation des paramètres d'une caméra
 - Déterminer les transformations subies par un point de l'espace pour obtenir un point de l'image
 - Modèle : Pin Hole¹ ou sténopé
 - Repère scène \Leftrightarrow repère caméra
 - Paramètres extrinsèques
 - Repère caméra \Leftrightarrow repère image
 - Paramètres intrinsèques



¹ Trou d'épingle

Calibration : Paramètres intrinsèques

T.H.

- Transformation perspective (eq. 1)
 - (x_c, y_c, z_c) : point (x, y, z) projeté dans le repère caméra
 - f : distance focale
- Transformation caméra / image (eq.2)
 - (u, v) : point image
 - (u_0, v_0) : axe optique dans l'image
 - (k_u, k_v) facteurs d'échelle repère métrique caméra \Leftrightarrow repère pixels image
- Paramètres intrinsèques (eq. 3)
 - $\alpha_u = k_u f, \alpha_v = k_v f, u_0, v_0$

équation 1 : Transformation perspective

$$\begin{pmatrix} sx_c \\ sy_c \\ sz_c \\ s \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{pmatrix}}_P \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \Leftrightarrow \begin{cases} \frac{x_c}{f} = \frac{x}{z} \\ \frac{y_c}{f} = \frac{y}{z} \\ z_c = f \\ s = \frac{z}{f} \end{cases}$$

\times

équation 3 : Transformation caméra / image (2)

$$\left. \begin{array}{l} u = \frac{\alpha_u x + u_0 z}{s} \\ v = \frac{\alpha_v y + v_0 z}{s} \\ s = z \end{array} \right\} \begin{pmatrix} su \\ sv \\ s \end{pmatrix} = \underbrace{\begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{I_c} \begin{pmatrix} x \\ y \\ z \\ s \end{pmatrix}$$

équation 2 : Transformation caméra / image (1)

$$\Leftrightarrow \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \underbrace{\begin{pmatrix} k_u & 0 & 0 & u_0 \\ 0 & k_v & 0 & v_0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_K \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} \left\{ \begin{array}{l} u = k_u x_c + u_0 \\ v = k_v y_c + v_0 \\ w = 1 \end{array} \right.$$

Distortions

- Entre le modèle de projection linéaire et les points relevés dans l'image
- Sources :
 - Imperfection des lentilles
 - Mauvais alignement lentilles / matrice
- Formes
 - en g^{al} Radiales >> Tangentielles
 - $(\Delta u, \Delta v)$ peuvent être hautement non linéaires si l'on prends en compte les distorsions radiales et tangentielles
- Prise en compte
 - K_1 rajoute un paramètre au modèle caméra et détruit la linéarité du modèle
 - Estimation de la distorsion indépendamment du modèle et rectification des images
 - On travaillera alors sur les pixels corrigés (u', v') : $\begin{cases} u' = u - \Delta u(u, v) \\ v' = v - \Delta v(u, v) \end{cases}$



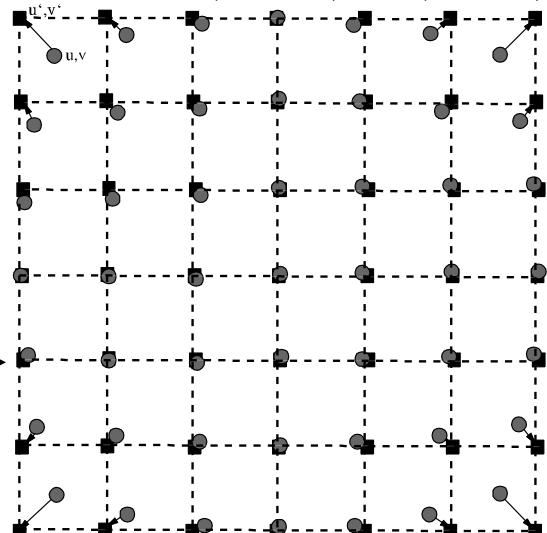
équation 4 : distorsions

$$\begin{cases} u = k_u x_c + u_0 + \Delta u \\ v = k_v y_c + v_0 + \Delta v \end{cases}$$

équation 5 : distorsion radiale

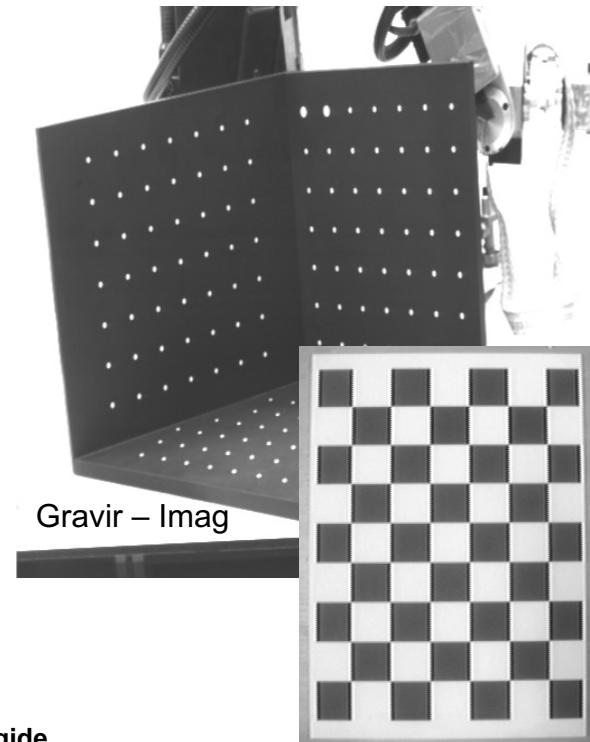
$$\begin{cases} \Delta u = K_1 r^2 (u - u_0) \\ \Delta v = K_1 r^2 (v - v_0) \end{cases}$$

avec $r^2 = (u - u_0)^2 + (v - v_0)^2$



Paramètres extrinsèques

- Déplacement rigide entre le repère scène et le repère caméra
- Repère scène symbolisé par une mire de calibrage contenant des points 3D connus
- Estimer le déplacement rigide entre la mire et la caméra d'après le relevé des points particuliers dans les images



équation 6 : Matrice de déplacement rigide

$$A = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{r}_1 & t_x \\ \mathbf{r}_2 & t_y \\ \mathbf{r}_3 & t_z \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{pmatrix} \text{ avec } A^{-1} = \begin{pmatrix} \mathbf{R}^t & -\mathbf{R}^t \mathbf{t} \\ 0 & 1 \end{pmatrix}$$

Calibration avec OpenCV

$$sm = A(R|t)M$$

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix}}_{(R|t)} \underbrace{\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}}_{\begin{pmatrix} x & y & z \end{pmatrix}^t}$$

$m^t = \begin{pmatrix} u & v & 1 \end{pmatrix}$: point 2D image

$M^t = \begin{pmatrix} X & Y & Z & 1 \end{pmatrix}$: point 3D scène

$M_c^t = \begin{pmatrix} x & y & z \end{pmatrix}$: point 3D dans repère caméra

A = matrice caméra = paramètres intrinsèques

(f_x, f_y) = longueurs focales en pixels

(c_x, c_y) = point principal : axe optique dans l'image

$(R|t)$ = Rotation/Translation = paramètres extrinsèques

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + t$$

sans distorsions

$$x' = x / z$$

$$y' = y / z$$

$$u = f_x \cdot x' + c_x$$

$$v = f_y \cdot y' + c_y$$

avec distorsions

$$x' = x / z$$

$$y' = y / z$$

$$x'' = x' \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + 2 p_1 x' y' + p_2 (r^2 + 2x'^2)$$

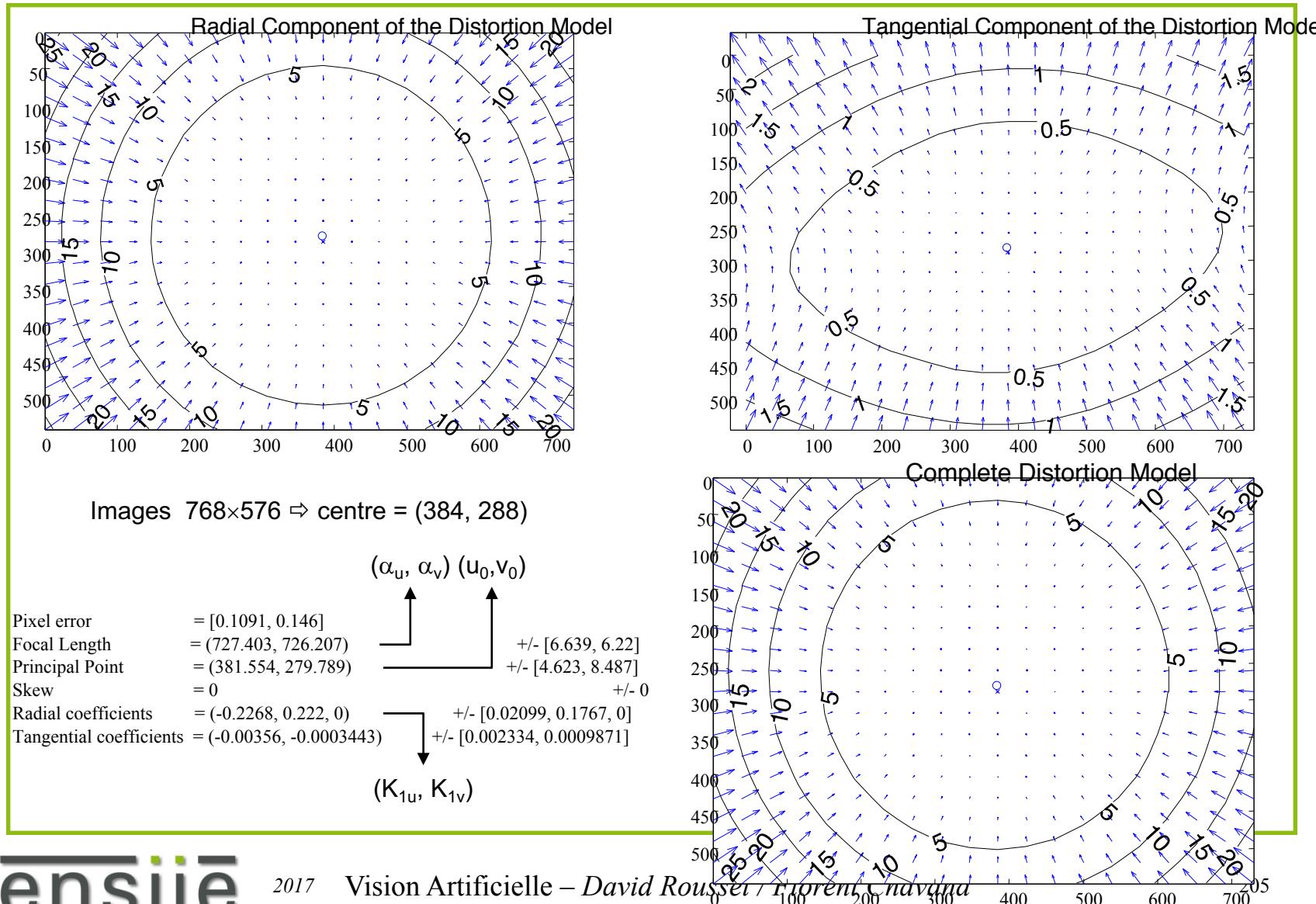
$$y'' = y' \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + 2 p_2 x' y' + p_1 (r^2 + 2y'^2)$$

$$\text{où } r^2 = x'^2 + y'^2$$

$$u = f_x \cdot x'' + c_x \quad [k_1 \dots k_6]: \text{Coefs distorsion radiale}$$

$$v = f_y \cdot y'' + c_y \quad [p_1, p_2]: \text{Coefs distorsion tangentielle}$$

Exemple de distorsions sur une webcam



Transformation Scène \Rightarrow Image

- La transformation complète subie par un point de l'espace de la scène pour produire un point dans le repère de l'image est donc décrite par le produit $I_c \cdot A$:

$$M = I_c A \quad \text{équation 7 : Transformation scène / image}$$

$$\begin{aligned}
 &= \begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
 &= \begin{pmatrix} \alpha_u r_1 + u_0 r_3 & \alpha_u t_x + u_0 t_z \\ \alpha_v r_2 + v_0 r_3 & \alpha_v t_y + v_0 t_z \\ r_3 & t_z \end{pmatrix} \\
 &= \begin{pmatrix} \alpha_u r_{11} + u_0 r_{31} & \alpha_u r_{12} + u_0 r_{32} & \alpha_u r_{13} + u_0 r_{33} & \alpha_u t_x + u_0 t_z \\ \alpha_v r_{21} + v_0 r_{31} & \alpha_v r_{22} + v_0 r_{32} & \alpha_v r_{23} + v_0 r_{33} & \alpha_v t_y + v_0 t_z \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix} \\
 &= \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix} = \begin{pmatrix} \mathbf{m}_1 & \mathbf{m}_{14} \\ \mathbf{m}_2 & \mathbf{m}_{24} \\ \mathbf{m}_3 & \mathbf{m}_{34} \end{pmatrix}
 \end{aligned}$$

avec

$$\begin{cases} \mathbf{r}_3 = \mathbf{m}_3 \\ u_0 = \mathbf{m}_1 \cdot \mathbf{m}_3 \\ v_0 = \mathbf{m}_2 \cdot \mathbf{m}_3 \\ \alpha_u = -\|\mathbf{m}_1 \wedge \mathbf{m}_3\| \\ \alpha_v = \|\mathbf{m}_2 \wedge \mathbf{m}_3\| \\ \mathbf{r}_1 = 1/\alpha_u (\mathbf{m}_1 - u_0 \mathbf{m}_3) \\ \mathbf{r}_2 = 1/\alpha_v (\mathbf{m}_2 - v_0 \mathbf{m}_3) \\ t_x = 1/\alpha_u (m_{14} - u_0 m_{34}) \\ t_y = 1/\alpha_v (m_{24} - v_0 m_{34}) \\ t_z = m_{34} \end{cases}$$

Estimation des paramètres (1/2)

$$\begin{pmatrix} su \\ sv \\ s \end{pmatrix} \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \Rightarrow$$

équation 8 : Points projetés dans l'image

$$\begin{cases} u = \frac{m_{11}X + m_{12}Y + m_{13}Z + m_{14}}{m_{31}X + m_{32}Y + m_{33}Z + m_{34}} \\ v = \frac{m_{21}X + m_{22}Y + m_{23}Z + m_{24}}{m_{31}X + m_{32}Y + m_{33}Z + m_{34}} \end{cases}$$

- 1 pt 3D \Rightarrow 2 (coords image|équations)

$$\begin{cases} X_i m_{11} + Y_i m_{12} + Z_i m_{13} + m_{14} - u_i X_i m_{31} - u_i Y_i m_{32} - u_i Z_i m_{33} = u_i m_{34} \\ X_i m_{21} + Y_i m_{22} + Z_i m_{23} + m_{24} - v_i X_i m_{31} - v_i Y_i m_{32} - v_i Z_i m_{33} = v_i m_{34} \end{cases}$$

\Rightarrow au moins 6 points pour estimer les 12 paramètres du modèle : $n > 6$ points \Rightarrow système de $2n$ équations

$(K_{2n \times 11})$

$(x_{11 \times 1}) = (u_{2n \times 1})$

- Contrainte : $t_z = m_{34} = 1 \Leftrightarrow$ on fixe arbitrairement la translation en Z scène / caméra
- Un tel système surcontraint (avec un nombre de points) issus de données bruitées ne pourra être résolu que par une approximation (au sens des moindres carrés par exemple)
- Dans un système non parfait on aura toujours une erreur résiduelle, aussi on pourra écrire l'équation 9 sous la forme $Kx - u = e$ où e représente un vecteur d'erreurs

$$\begin{pmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & -u_i X_i & -u_i Y_i & -u_i Z_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -v_i X_i & -v_i Y_i & -v_i Z_i \end{pmatrix}$$

$$\begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{34} \end{pmatrix} = \begin{pmatrix} \vdots \\ u_i m_{34} \\ v_i m_{34} \\ \vdots \end{pmatrix}$$

équation 9 : Estimation des paramètres de projection

Estimation des paramètres (2/2)

- Rechercher une solution qui minimise l'erreur « e »

- On cherche donc m_{ij} tel que $\|e\|^2 = e^t e$ soit minimum :

$$\begin{aligned} e^t e &= (Kx - u)^t (Kx - u) \\ &= x^t K^t K x - u^t K x - x^t K^t u + u^t u \\ &= x^t K^t K x - 2x^t K^t u + u^t u \end{aligned}$$

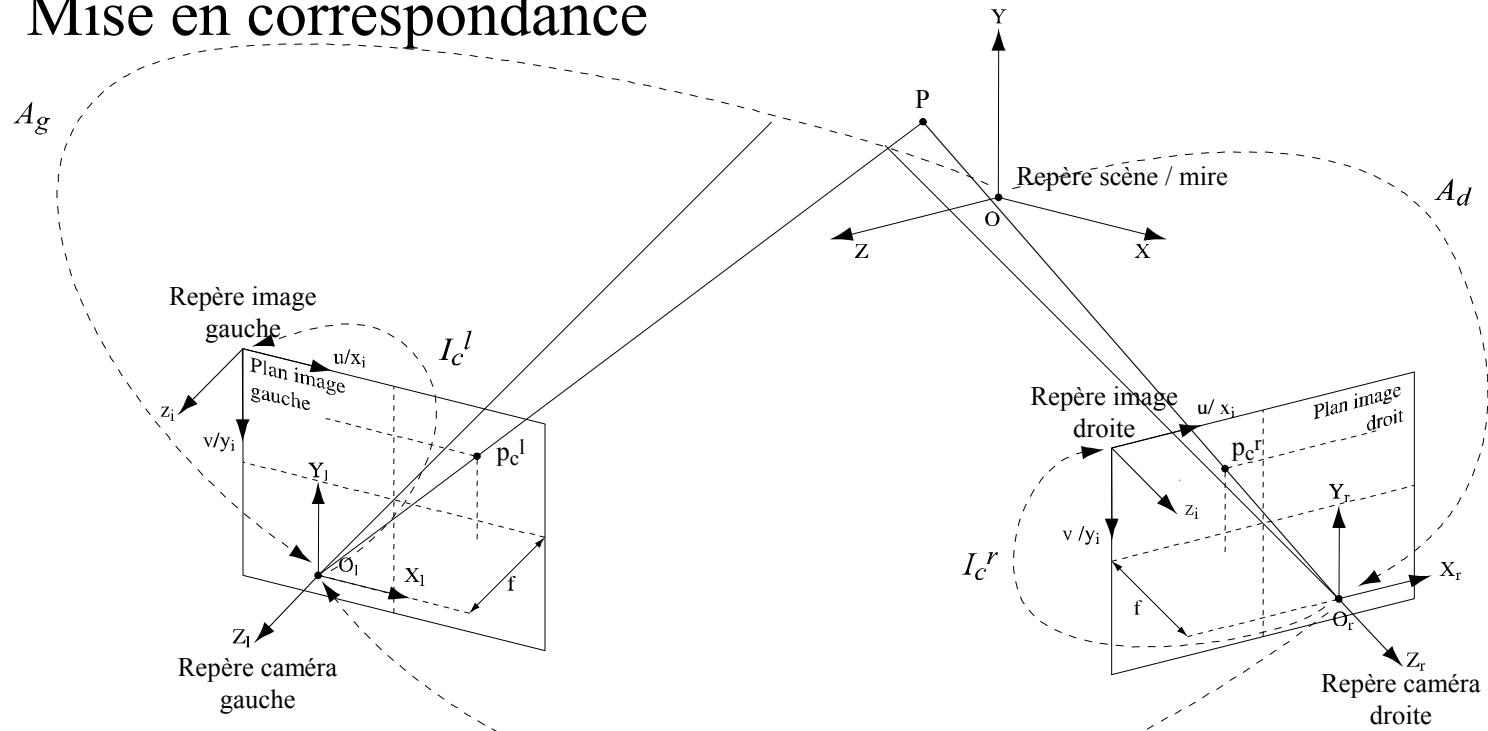
$$\frac{d(e^t e)}{dx} = K^t K x - K^t u = 0$$

$$x = (K^t K)^{-1} K^t u$$

- où $(K^t K)^{-1} K^t$ est la pseudo inverse de K et $(K^t K)^{-1}$ est une matrice 11×11 symétrique et positive qui peut être inversée par des techniques numériques comme
 - l'élimination de Gauss-Jordan,
 - la décomposition LU ou QR
 - ou encore une décomposition en valeurs singulières.

Calibration stéréo

- Mise en correspondance



équation 10 : Transformation Caméra gauche / droite (1)

A_s

$$A_s = A_d A_g^{-1}$$

$$A_{\text{camera droite} \leftarrow \text{camera gauche}} = A_{\text{camera droite} \leftarrow \text{mire}} A_{\text{mire} \leftarrow \text{camera gauche}}$$

Calibration stéréo : Géométrie épipolaire (1/2)

- But : Déterminer la géométrie épipolaire de la paire stéréo
 - Mouvement entre les deux caméras
- Mises en correspondance :
 - Relations entre un point de l'image gauche $p=(x,y,1)$ et un point de l'image droite $p'=(x',y',1)$
 - Contrainte épipolaire
 - Point dans une image \Rightarrow lieu où doit se trouver son correspondant dans l'autre image = droite épipolaire

équation 11 : Transformation Caméra gauche / droite (2)

$$A_s = \begin{pmatrix} r_{11} & r_{12} & r_{13} & b_x \\ r_{21} & r_{22} & r_{23} & b_y \\ r_{31} & r_{32} & r_{33} & b_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} r_1 & b_x \\ r_2 & b_y \\ r_3 & b_z \\ 0 & 1 \end{pmatrix}$$

équation 12 : Expression d'un point de l'image droite / l'image gauche

$$x' = \frac{Zr_1 \cdot p + b_x}{Zr_3 \cdot p + b_z}$$

$$y' = \frac{Zr_2 \cdot p + b_y}{Zr_3 \cdot p + b_z}$$

équation 13 : Paramètres d'une droite épipolaire
 $ax' + by' + c = 0$

$$a = (b_y r_{31} - b_z r_{21})x + (b_y r_{32} - b_z r_{22})y + (b_y r_{33} - b_z r_{23})$$

$$b = (b_z r_{11} - b_x r_{31})x + (b_z r_{12} - b_x r_{32})y + (b_z r_{13} - b_x r_{33})$$

$$c = (b_x r_{21} - b_y r_{11})x + (b_x r_{22} - b_y r_{12})y + (b_x r_{23} - b_y r_{13})$$

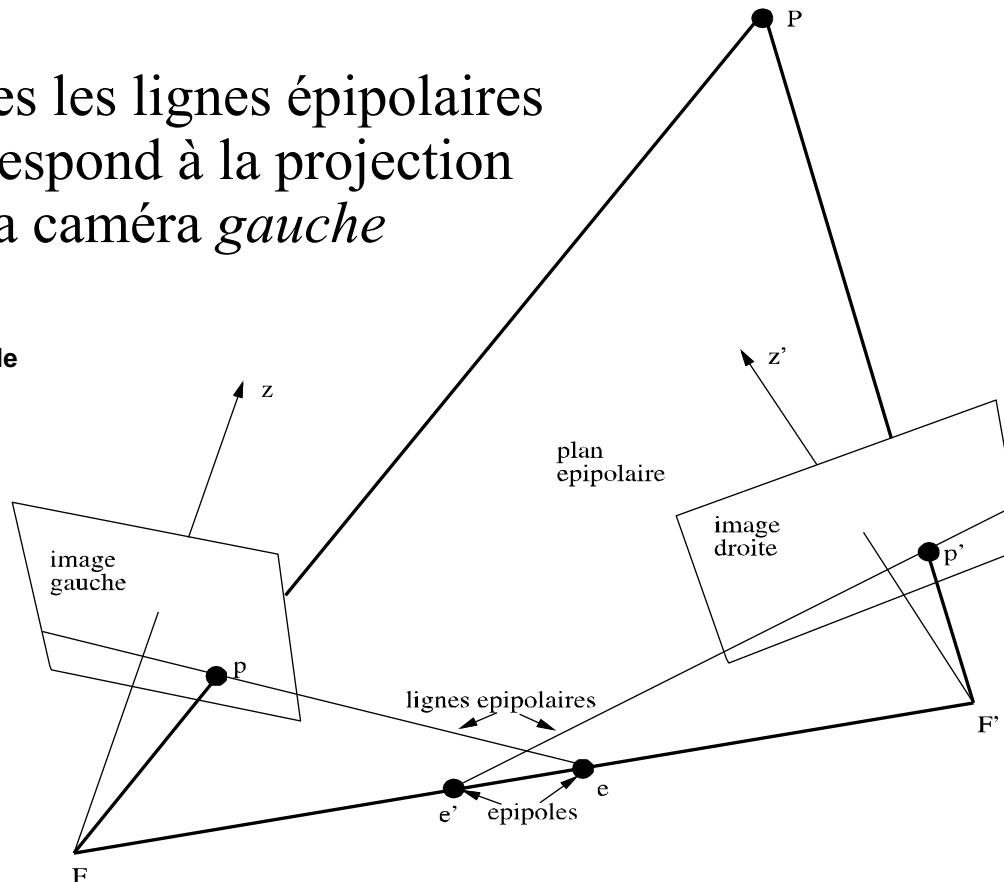
Calibration stéréo : Géométrie épipolaire (2/2)

- Ligne épipolaire *droite*: lieu le long duquel on doit chercher un correspondant à un point dans le repère de l'image *gauche* et vice versa.
- L'intersection de toutes les lignes épipolaires de l'image *droite* correspond à la projection du centre optique de la caméra *gauche* dans le repère droit

équation 14 : Equation de l'épiphore

$$\begin{cases} x'_e = \frac{b_x}{b_z} \\ y'_e = \frac{b_y}{b_z} \end{cases}$$

- Cas particulier
 - Axes optiques //
 - Ligne épipolaires horizontales



Calibration stéréo : Matrices E & F

- Matrice Essentielle : relations points caméras

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \underbrace{\begin{pmatrix} 0 & -b_z & b_y \\ b_z & 0 & b_x \\ -b_y & b_x & 0 \end{pmatrix}}_{E} \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

équation 15 : Matrice Essentielle E

équation 16 : relation entre les points d'une paire d'images dans les repères caméras

$$\begin{pmatrix} x' & y' & 1 \end{pmatrix} E \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0 \Leftrightarrow p'^t E p = 0$$

- Matrice Fondamentale : relations points images

équation 17 : Transformation caméra / image, version réduite

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$m = C \cdot p$

– Propriétés :

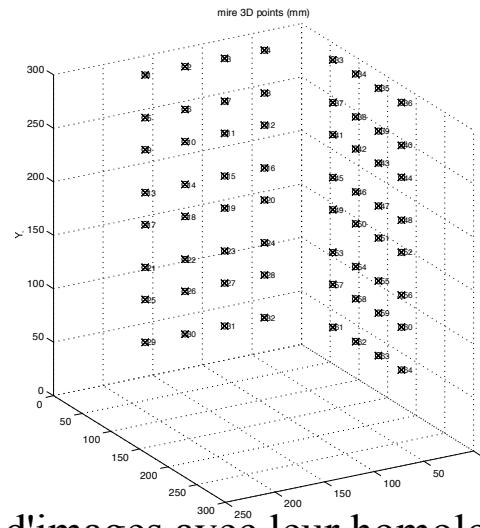
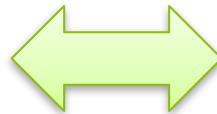
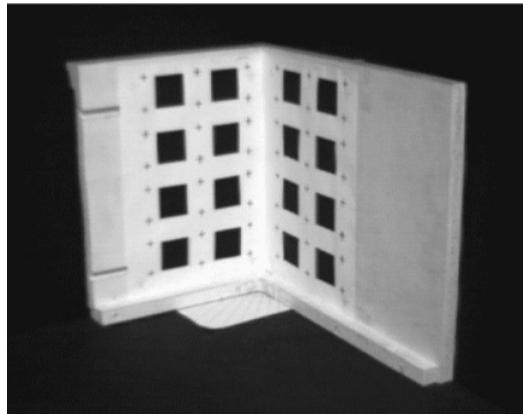
- Transposée : Si F est la matrice fondamentale d'une paire de projections (M, M') alors F^t est la matrice fondamentale de la paire (M', M) .
- Lignes épipolaires : pour tout point x d'une image d'une paire, la ligne épipolaire dans l'autre image est obtenue par $l = Fx$. Inversement $l = F^tx'$ représente la ligne épipolaire correspondant à x' dans la première image
- Epipôle : pour tout point x (autre que l'épipôle e) la ligne épipolaire $l = Fx$ contient l'épipôle e' . D'où e' vérifie $e'^t(Fx) = (e'^tF)x = 0$ pour tout x , et donc $e'^tF = \mathbf{0}$ ou encore $F^te' = \mathbf{0}$. De même $Fe = \mathbf{0}$

équation 18 : Matrice Fondamentale F à partir de la matrice essentielle

$$\underbrace{\begin{pmatrix} u' & v' & 1 \end{pmatrix} (C'^{-1}) EC^{-1}}_F \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = 0$$

Calibration : approche expérimentale (1/2)

- Mire de calibrage



- Apparier des couples de points dans la paire d'images avec leur homologue tridimensionnel.
- Les points choisis pour cet appariement seront dans notre cas les coins de chaque carré noir de la mire, ce qui nous donne un total de 64 points utilisables pour la calibration
- Estimation des paramètres
 - L'équation 9 requiert au moins 6 points pour être résolue en fixant arbitrairement $m_{34} = 1 \Rightarrow$ pas de reconstruction métrique
 - Approche différente « Faugeras-Toscani » : si on pouvait déterminer M ($\text{mire} \rightarrow \text{image}$) par des mesures physiques on aurait alors :

$$\|m_3\|^2 = m_{31}^2 + m_{32}^2 + m_{33}^2 = 1 \quad \text{équation 19 : Contrainte de Faugeras-Toscani}$$

Estimation des paramètres méthode « Faugeras-Toscani » (1/2)

- Reécriture de l'équation 6 avec la contrainte de « Faugeras-Toscani »

- On recherche $x_9 = \begin{pmatrix} m_1 & m_{14} & \overset{\vee}{m}_2 & m_{24} & m_{34} \end{pmatrix}^t$ et $x_3 = (m_3)^t$ en minimisant le critère

$$Q = \left\| B_{(2n \times 9)X_9} + C_{(2n \times 3)X_3} \right\|^2 \quad \text{équation 21 : minimisation Faugeras-Toscani}$$

Minimiser Q revient alors à trouver les zéros de ses dérivées partielles suivant x_3 et x_9 .

$$\begin{cases} B^T B x_9 + B^T C x_3 = 0 \\ C^T C x_3 + C^T B x_9 - \lambda x_3 = 0 \end{cases}$$

**équation 23 : Faugeras-Toscani minimisation
dérivées partielles**

Estimation des paramètres méthode « Faugeras-Toscani » (2/2)

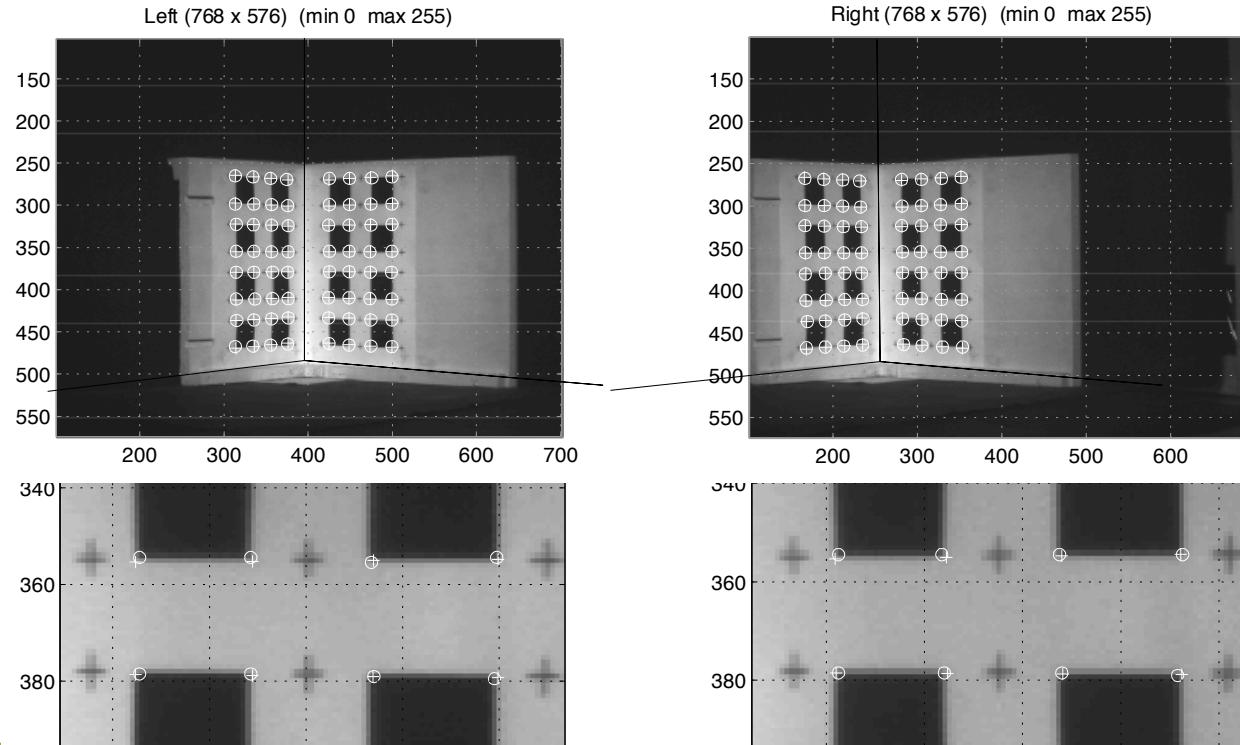
$$x_9 = -(\mathbf{B}^t \mathbf{B})^{-1} \mathbf{B}^t \mathbf{C} \mathbf{x}_3$$
$$\underbrace{(\mathbf{C}^t \mathbf{C} - \mathbf{C}^t \mathbf{B} (\mathbf{B}^t \mathbf{B})^{-1} \mathbf{B}^t \mathbf{C})}_{\mathbf{D}} \mathbf{x}_3 = \lambda \mathbf{x}_3$$

équation 24 : Faugeras-Toscani résolution

- D est une matrice 3×3 symétrique positive, elle a donc des valeurs propres réelles et positives. \mathbf{x}_3 est déterminé par le vecteur propre associé à la plus petite valeur propre λ de D.
- La détermination des paramètres m_{ij} doit donc suivre l'algorithme suivant :
 1. Calculer les valeurs propres de D
 2. Choisir la plus petite des valeurs propres λ pour minimiser le critère
 3. Obtenir le vecteur propre qui lui est associé, puis le normaliser pour ainsi obtenir \mathbf{x}_3 avec $\|\mathbf{x}_3\|^2 = 1$
 4. Puis calculer \mathbf{x}_9 à partir de l'équation 24
 5. La matrice M pour chaque caméra est donc composée des valeurs extraites de \mathbf{x}_3 et \mathbf{x}_9 . La contrainte utilisée (équation 19) définit M au signe près. Il faut donc décider du signe en utilisant le fait que l'objet de calibrage se situe devant ou derrière le repère des caméras : $m_{34} = t_z < 0$

Calibration : approche expérimentale (2/2)

- Résultat sur une paire d'images stéréo
 - \circ : points fournis pour la calibration,
 - $+$: points issus de la reprojection des points 3D en utilisant les matrices de transformation déterminées par la calibration



Détermination de la matrice fondamentale

- si l'on dispose d'un ensemble de correspondances entre deux images $x_i \leftrightarrow x'_i$ avec $i \in [1 \dots n]$ correspondant à un ensemble de points 3D X_i alors : $x_i^t F x_i = 0$

$$x^t F x = 0$$

$$(x' \ y' \ 1) \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0$$

équation 25 : matrice fondamentale (détail)

$$\Rightarrow x' x f_{11} + x' y f_{12} + x' f_{13} + y' x f_{21} + y' y f_{22} + y' f_{23} + x f_{31} + y f_{32} + f_{33} = 0$$

- Hartley & Zisserman proposent une méthode utilisant la matrice fondamentale et permettant à partir de 2 vues de :

- Calculer la matrice fondamentale à partir de correspondances de points dans la paire d'image
- Calculer les matrices des caméras à partir de la matrice fondamentale \Leftrightarrow Autocalibration
- Pour chaque correspondance de points $x_i \leftrightarrow x'_i$, calculer le point de l'espace X_i se projetant sur ces deux points images \Leftrightarrow Reconstruction
- Une variante de cette méthode consiste à utiliser la matrice essentielle si les caméras sont déjà calibrées, puisqu'on peut dans ce cas exprimer les correspondances dans les repères liés aux deux caméras : $(x_i^c)^t E x_i^c = 0$
- f est déterminée à un facteur près (eq. homogènes), le rang de A doit donc être ≥ 8 , on cherche alors une solution qui minimise $\|Af\|$ au sens des moindres carrés.
 - Vecteur propre de la SVD de A associé à la plus petite valeur propre.

$$\underbrace{\begin{pmatrix} x_1' x_1 & x_1' y_1 & x_1' & y_1' x_1 & y_1' y_1 & y_1' & x_1 & y_1 & 1 \\ \vdots & \vdots \\ x_n' x_n & x_n' y_n & x_n' & y_n' x_n & y_n' y_n & y_n' & x_n & y_n & 1 \end{pmatrix}}_A = 0$$

$$\underbrace{\begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{pmatrix}}_f$$

Reconstruction stéréo

équation 27 : projection d'un point P dans une paire d'images

- En appliquant l'équation 5 au cas stéréo, on obtient : \rightarrow
- (u_1, v_1) point image gauche, (u_2, v_2) point image droite
- (X, Y, Z) pt 3D reconstruit
- D'où l'on tire un système d'équations linéaires $A_{4 \times 3} X_{3 \times 1} = b_{4 \times 1}$

$$\begin{pmatrix} m_{11}^1 - u_1 m_{31}^1 & m_{12}^1 - u_1 m_{32}^1 & m_{13}^1 - u_1 m_{33}^1 \\ m_{21}^1 - v_1 m_{31}^1 & m_{22}^1 - v_1 m_{32}^1 & m_{23}^1 - v_1 m_{33}^1 \\ m_{11}^2 - u_2 m_{31}^2 & m_{12}^2 - u_2 m_{32}^2 & m_{13}^2 - u_2 m_{33}^2 \\ m_{21}^2 - v_2 m_{31}^2 & m_{22}^2 - v_2 m_{32}^2 & m_{23}^2 - v_2 m_{33}^2 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} u_1 m_{34}^1 - m_{14}^1 \\ v_1 m_{34}^1 - m_{24}^1 \\ u_2 m_{34}^2 - m_{14}^2 \\ v_2 m_{34}^2 - m_{24}^2 \end{pmatrix}$$

$$\left\{ \begin{array}{l} u_1 = \frac{m_{11}^1 X + m_{12}^1 Y + m_{13}^1 Z + m_{14}^1}{m_{31}^1 X + m_{32}^1 Y + m_{33}^1 Z + m_{34}^1} \\ v_1 = \frac{m_{21}^1 X + m_{22}^1 Y + m_{23}^1 Z + m_{24}^1}{m_{31}^1 X + m_{32}^1 Y + m_{33}^1 Z + m_{34}^1} \\ u_2 = \frac{m_{11}^2 X + m_{12}^2 Y + m_{13}^2 Z + m_{14}^2}{m_{31}^2 X + m_{32}^2 Y + m_{33}^2 Z + m_{34}^2} \\ v_2 = \frac{m_{21}^2 X + m_{22}^2 Y + m_{23}^2 Z + m_{24}^2}{m_{31}^2 X + m_{32}^2 Y + m_{33}^2 Z + m_{34}^2} \end{array} \right.$$

équation 28 : reconstruction d'un point P à partir des points images p1 & p2

- Inversion (au sens de la pseudo inverse par SVD) : $X_{3 \times 1} = A_{3 \times 4}^{-1} b_{4 \times 1}$

- Reconstruction dans l'un ou l'autre des repères caméras (équation 12) :

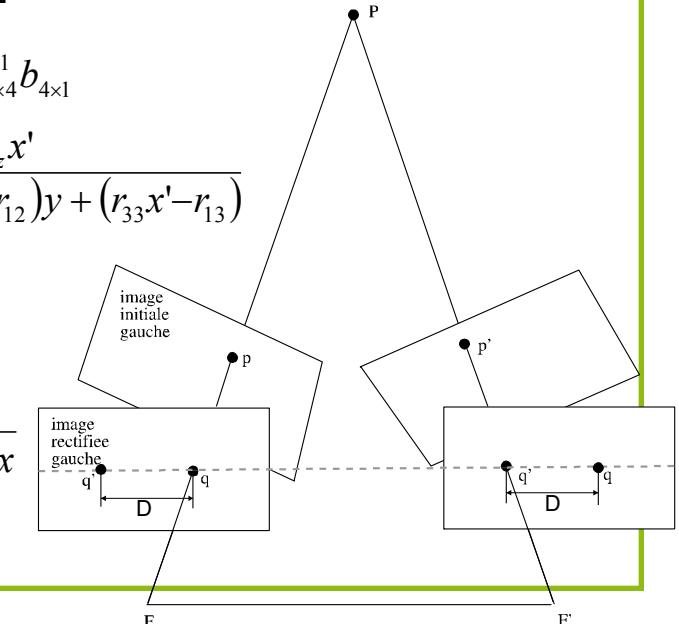
$$x' = \frac{(r_{11}x + r_{12}y + r_{13})Z + b_x}{(r_{31}x + r_{32}y + r_{33})Z + b_z} \Rightarrow$$

- Cas particulier des axes optiques parallèles ou images rectifiées :

- $D = (x' - x)$ = disparité des pts appariés
- les lignes épipolaires sont parallèles et horizontales et les épipôles sont rejettés à l'infini

$$\left\{ \begin{array}{l} Z = \frac{b_x - b_z x'}{(r_{31}x' - r_{11})x + (r_{32}x' - r_{12})y + (r_{33}x' - r_{13})} \\ X = xZ \\ Y = yZ \end{array} \right.$$

$$\left\{ \begin{array}{l} x' = \frac{Zx + b}{Z} \Rightarrow \\ y' = y \end{array} \right\} \left\{ \begin{array}{l} Z = \frac{b}{x' - x} \\ X = xZ \\ Y = yZ \end{array} \right.$$



Reconstruction N vues

- Généralisation de l'équation 28 : $n \geq 2$

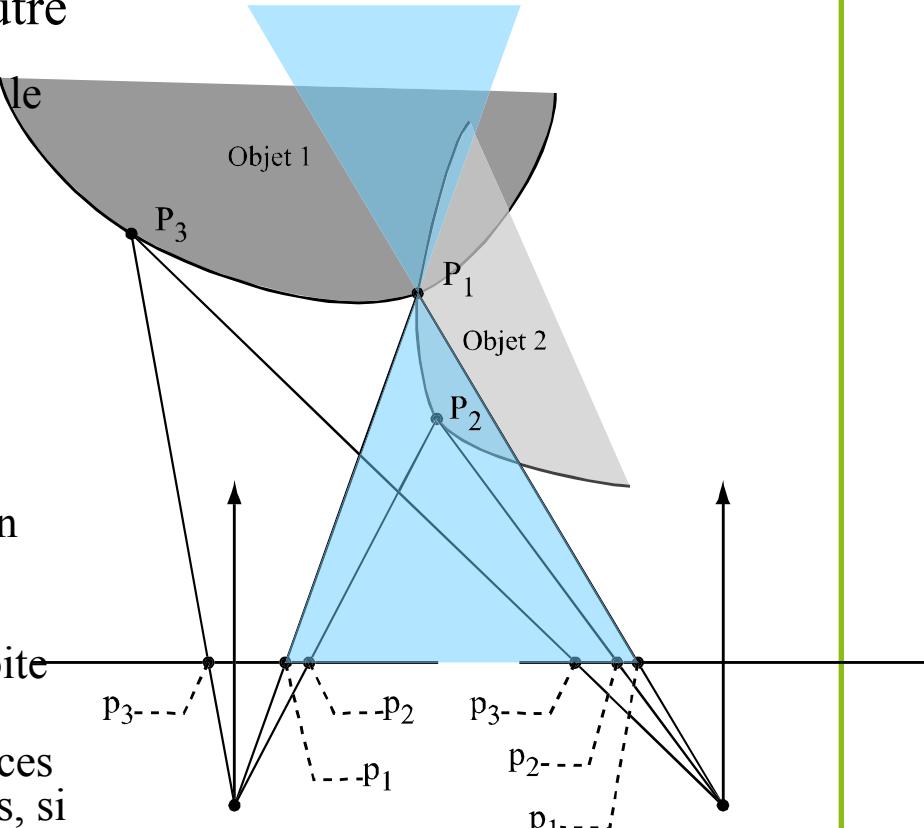
$$\begin{pmatrix} m_{11}^1 - u_1 m_{31}^1 & m_{12}^1 - u_1 m_{32}^1 & m_{13}^1 - u_1 m_{33}^1 \\ m_{21}^1 - v_1 m_{31}^1 & m_{22}^1 - v_1 m_{32}^1 & m_{23}^1 - v_1 m_{33}^1 \\ \vdots & \vdots & \vdots \\ m_{11}^n - u_n m_{31}^n & m_{12}^n - u_n m_{32}^n & m_{13}^n - u_n m_{33}^n \\ m_{21}^n - v_n m_{31}^n & m_{22}^n - v_n m_{32}^n & m_{23}^n - v_n m_{33}^n \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} u_1 m_{34}^1 - m_{14}^1 \\ v_1 m_{34}^1 - m_{24}^1 \\ \vdots \\ u_n m_{34}^n - m_{14}^n \\ v_n m_{34}^n - m_{24}^n \end{pmatrix}$$
$$A_{2n \times 3} \cdot X_{3 \times 1} = b_{2n \times 1}$$

Contraintes pour la mise en correspondance

- Mise en correspondance automatique de points entre images
 - Les points appariés (ou tout autre primitive) possèdent une ressemblance (au moins locale) d'une image à une autre
 - Corrélation de voisinages par exemple
 - Les contraintes issues de la géométrie épipolaire permettent de réduire l'espace de recherche de correspondant et de limiter ainsi le taux de faux appariements
- Contraintes classiques
 - Contrainte d'Ordre
 - Qui implique la contrainte d'Unicité
 - Limite du gradient de disparité
- Mise en correspondance hiérarchique

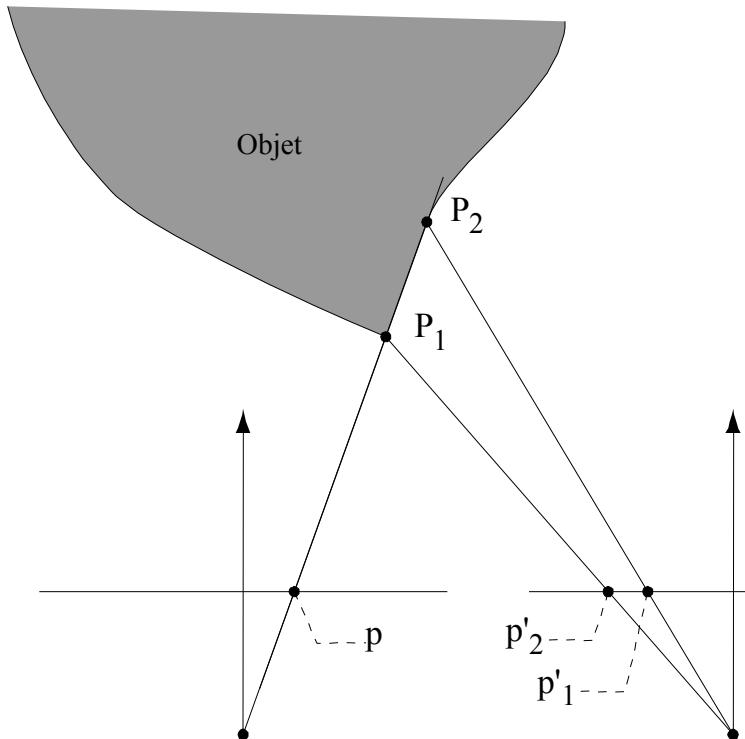
Contrainte d'ordre

- L'ordre des points projetés doit rester le même d'une image à l'autre
 - Les projections de P_1 partitionnent le plan épipolaire en deux régions : R_{externe} et R_{interne}
 - $P_3 \in R_{\text{externe}} \& P_2 \in R_{\text{interne}}$
 - Inversion d'ordre $(p_3, p_1, p_2)_g \leftrightarrow (p_3, p_2, p_1)_d$
 - $P_3 \in \text{Objet}_1$ peu incliné / plan des images
 - $P_2 \in \text{Objet}_2$ fortement incliné / plan des images
 - Si Objet_2 était opaque, P_2 n'apparaîtrait pas dans l'image droite
 - Cette situation n'intervient que lorsque la scène présente des surfaces transparentes et fortement inclinées, si ce n'est pas le cas on peut alors garantir la contrainte d'ordre



Contrainte d'unicité

- Chaque point 3D a une projection distincte dans les images
 - Cas limite de la contrainte d'ordre
 - La contrainte d'ordre ainsi que la contrainte d'unicité peuvent être exprimées analytiquement au travers de la limite du gradient de disparité



Limite du gradient de disparité

- disparité D $Z = \frac{b}{D} = \frac{b}{x' - x}$
- Gradient de disparité entre deux points P & Q

équation 30 : gradient de disparité

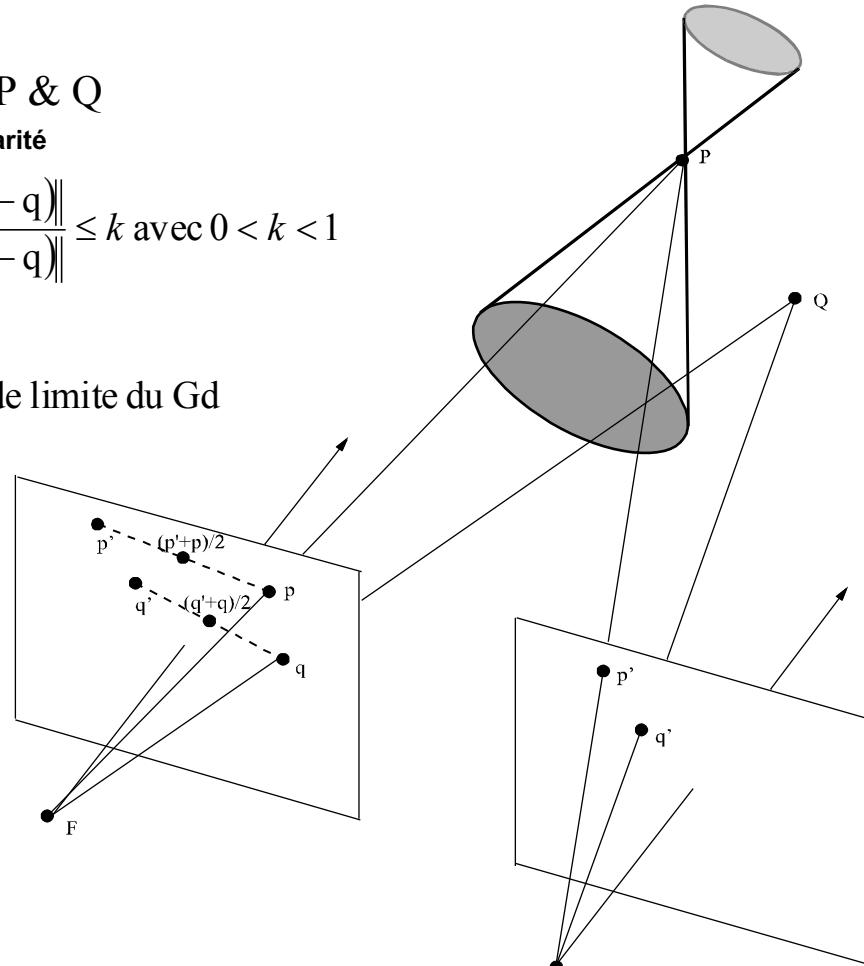
$$G_d = \frac{\|(p'-p) - (q'-q)\|}{\left\| \frac{(p'+p)}{2} + \frac{(q'+q)}{2} \right\|} \Rightarrow \frac{G_d}{2} = \frac{\|(p'-q') - (p-q)\|}{\|(p'-q') + (p-q)\|} \leq k \text{ avec } 0 < k < 1$$

$$\Rightarrow \frac{\|p'-q'\|}{\|p-q\|} < \frac{1+k}{1-k} = \frac{1}{\lambda} \Rightarrow \frac{\|p'-q'\|}{\|p-q\|} = \frac{1}{\lambda} = \text{lieu de limite du Gd}$$

- Les appariements (p, p') et (q, q') sont compatibles ssi leur $Gd < 2$
- Lieu de Q limite du Gd avec
 $p = (x_1 \quad y_1)^t$, $p' = (x'_1 \quad y_1)^t$,
 $q = (x_2 \quad y_2)^t$, $q' = (x'_2 \quad y_2)^t$
 – Cône centré en P :

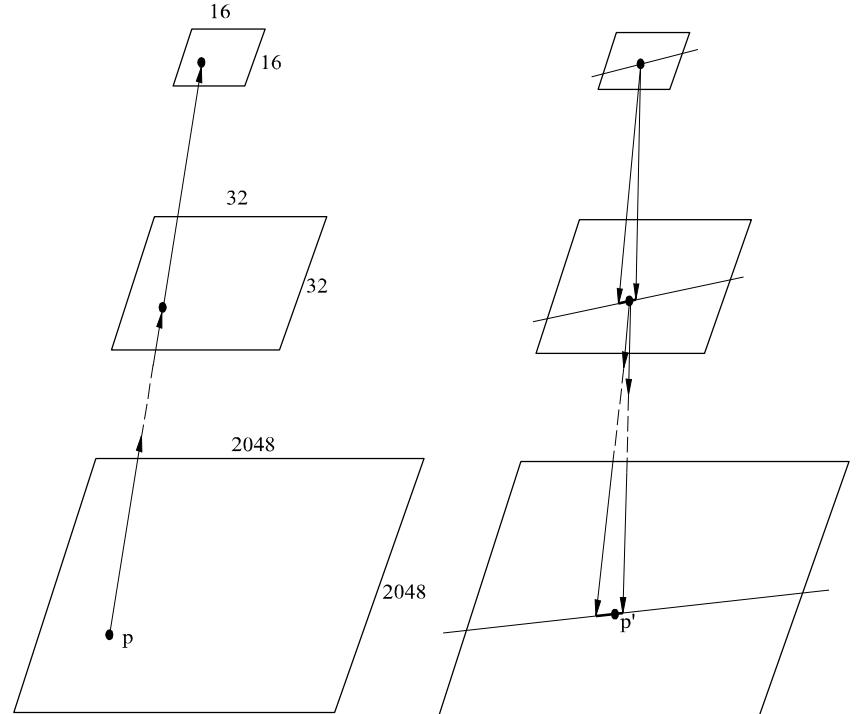
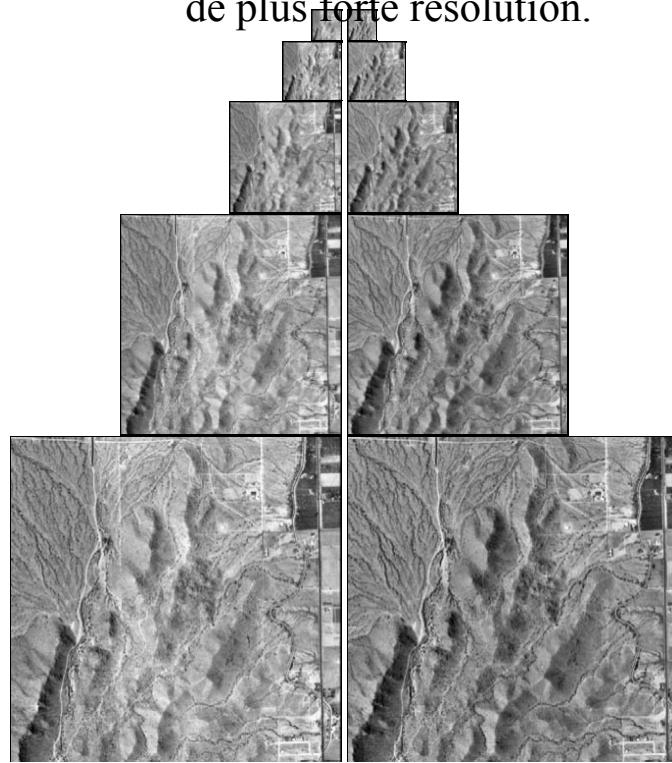
$$\begin{cases} y_2 = Y/Z \\ x_2 = X/Z \\ x'_2 = x_2 + b/Z \end{cases} \rightarrow \frac{(x_1 Z - X)^2 + (y_1 Z - Y)^2}{(x'_1 Z - (b+X))^2 + (y'_1 Z - Y)^2} = \lambda^2$$

$$(1 - \lambda^2)(X^2 + Y^2) + (x_1^2 + y_1^2 - \lambda^2(x_1^2 + y_1^2))Z^2 + 2(\lambda^2 - 1)x_1 X Z + 2(\lambda^2 y'_1 - y_1)YZ - 2\lambda^2 b Y + 2\lambda^2 y'_1 b Z - \lambda^2 b^2 = 0$$



Mise en correspondance hiérarchique

- Si une mise en correspondance est obtenue dans une paire d'images de résolution réduite.
- Cette mise en correspondance peut être utilisé pour initialiser une mise en correspondance dans une paire d'image de résolution plus importante
 - réduire l'espace de recherche des candidats à l'appariement dans la paire d'image de plus forte résolution.



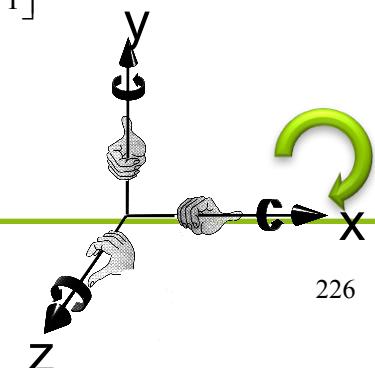
Bibliographie

- Radu Horaud & Olivier Monga - *Vision par ordinateur : outils fondamentaux* (2e édition revue et augmentée). Traité des Nouvelles Technologies, série informatique. Ed. Hermès, 1993, ISBN : 2-86601-481-2,
<http://perception.inrialpes.fr/people/Horaud/livre-fichiersPS/VO-HoraudMonga.pdf>
- Olivier Faugeras - *Three-dimensional computer vision: a geometric viewpoint*. Ed. MIT Press, 1993, ISBN : 0-262-06158-9
- R. I. Hartley & A. Zisserman - *Multiple View Geometry in Computer Vision* (Second). Ed. Cambridge University Press, 2004, ISBN : 0521540518

Matrices de transformations homogènes

- Translation : $T(d_x, d_y, d_z) = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Facteur d'échelle : $S(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Rotations : $R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $R_y(\varphi) = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\varphi & 0 & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $R_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 & 0 \\ \sin\psi & \cos\psi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
- Cisaillement : $SH_{xy}(sh_x, sh_y) = \begin{bmatrix} 1 & 0 & sh_x & 0 \\ 0 & 1 & sh_y & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$$\begin{bmatrix} x_t^h \\ y_t^h \\ z_t^h \\ w \end{bmatrix} = T_{4 \times 4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} x_t \\ y_t \\ z_t \\ 1 \end{bmatrix} = \frac{1}{w} \underbrace{\begin{bmatrix} x_t^h \\ y_t^h \\ z_t^h \\ w \end{bmatrix}}_{\text{normalisation}}$$



Transformation homogènes (2/2)

- Compositions : $\text{Tr} = \mathbf{R} \bullet \mathbf{S} \bullet \mathbf{T} \Leftrightarrow$ translation, puis facteur d'échelle, puis rotation
- Projections

– Perspective

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{d} & 0 \end{pmatrix}$$

– Orthographique

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Projection générique

$$\begin{pmatrix} 1 & 0 & -\frac{d_x}{d_z} & z_p \frac{d_x}{d_z} \\ 0 & 1 & -\frac{d_y}{d_z} & z_p \frac{d_y}{d_z} \\ 0 & 0 & -\frac{z_p}{Qd_z} & \frac{z_p^2}{Qd_z} + z_p \\ 0 & 0 & -\frac{1}{Qd_z} & \frac{z_p}{Qd_z} + 1 \end{pmatrix}$$

Projection	z_p	Q	d_x	d_y	d_z
Orthographique	0	∞	0	0	-1
Perspective	d	d	0	0	-1
Perspective (autre)	0	d	0	0	-1
Cavalière	0	∞	$\sin(\alpha)$	$\cos(\alpha)$	-1
Cabinet	0	∞	$\sin(\alpha)/2$	$\cos(\alpha)/2$	-1

Décomposition en valeurs singulières (SVD)

- Décomposition

- Toute matrice $A_{m \times n}$ avec $m > n$ peut être décomposée en un produit de matrices $M = U \cdot S \cdot V^t$ avec $U_{m \times m}$ une matrice unitaire, $S_{m \times n}$ la matrice diagonale dont les coefficients diagonaux sont des réels positifs ou nuls contenant les « valeurs propres », et V^t est la matrice adjointe à $V_{n \times n}$, matrice unitaire dont les colonnes contiennent les « vecteurs propres ».

$$\left(\begin{array}{c} M_{m \times n} \\ m \geq n \end{array} \right) = \underbrace{\left(\begin{array}{c} \\ \\ U_{m \times n} \\ \hline \end{array} \right)}_{\text{column orthogonal}} \cdot \underbrace{\left(\begin{array}{ccc} s_1 & & \\ & \ddots & \\ 0 & 0 & s_n \\ \vdots & & \end{array} \right)}_{\substack{S_{m \times n} : \text{diagonal} \\ \text{valeurs propres}}} \cdot \underbrace{\left(\begin{array}{c} \\ \\ V^t_{n \times n} \\ \hline \end{array} \right)}_{\substack{\text{orthogonal} = \\ \text{vecteurs propres}}} \quad s_1 > \dots > s_n$$

$$\begin{aligned} U \cdot U^t &= V \cdot V^t = I \\ \Rightarrow U^{-1} &= U^t \quad \& \quad V^{-1} = V^t \\ M_{m \times n} &= U_{m \times m} S_{m \times n} V^t_{n \times n} \\ \Rightarrow M_{m \times n} V_{n \times n} &= U_{m \times m} S_{m \times n} \\ \Rightarrow M_{m \times n} v_{n \times 1}^i &= u_{m \times 1}^i s_i \\ i \in [1 \dots n] \end{aligned}$$

- Inversion par SVD

$$\left(\begin{array}{c} M^{-1}_{n \times m} \end{array} \right) = \left(\begin{array}{c} V_{n \times n} \end{array} \right) \cdot \left(\begin{array}{ccc} \frac{1}{s_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{s_n} \end{array} \right) \cdot \left(\begin{array}{c} \\ \\ U^t_{n \times m} \\ \hline \end{array} \right)$$

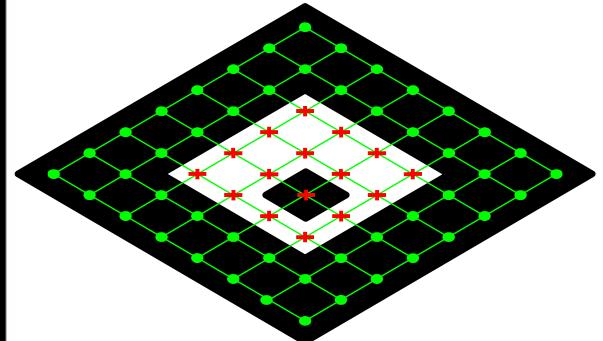
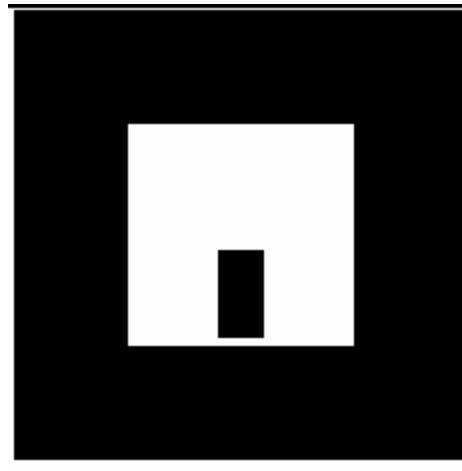
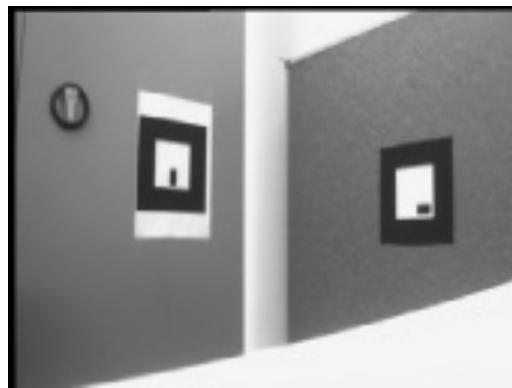
130

135

150

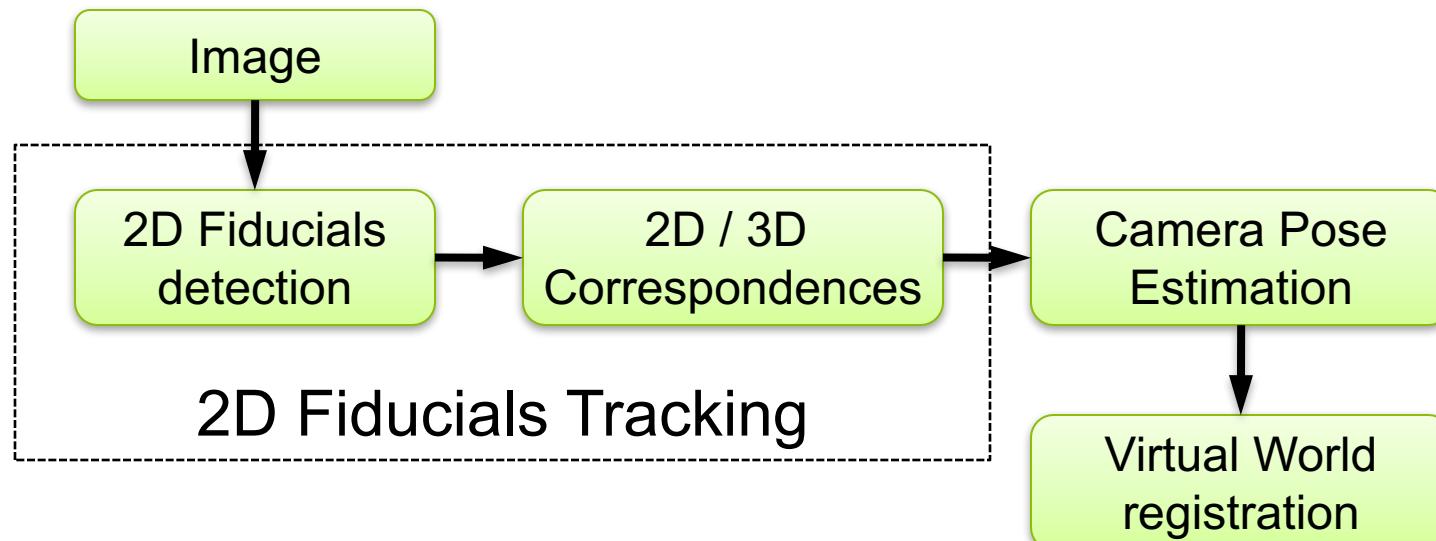
Etude de cas : Recalage / cible pour augmentation (1/5)

- Déterminer la pose de la caméra / à une cible
 - Déterminer la transformation rigide qui lie le repère scène R_m au repère caméra $R_c \Leftrightarrow$ Paramètres extrinsèques
- Reconnaître le contenu de la cible
- Augmenter la cible avec des éléments virtuels
 - Augmenter le monde réel \Leftrightarrow Réalité Augmentée
- Fakhr-eddine Ababsa & Malik Mallem , 2004. *Robust camera pose estimation using 2d fiducials tracking for real-time augmented reality systems*. VRCAI '04: Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry, ACM Press, pp. 431-435



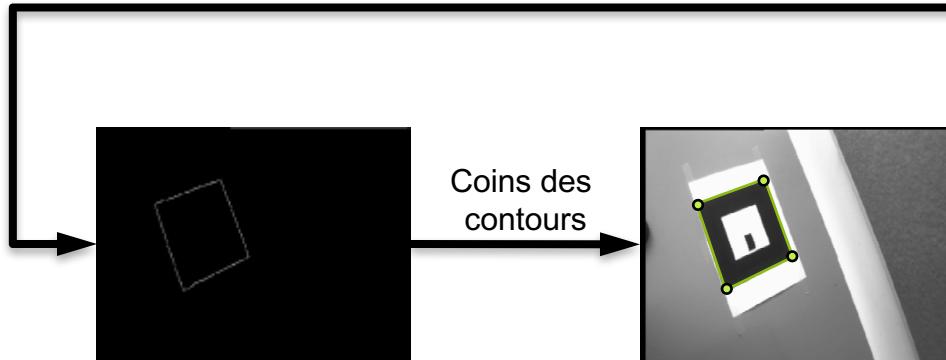
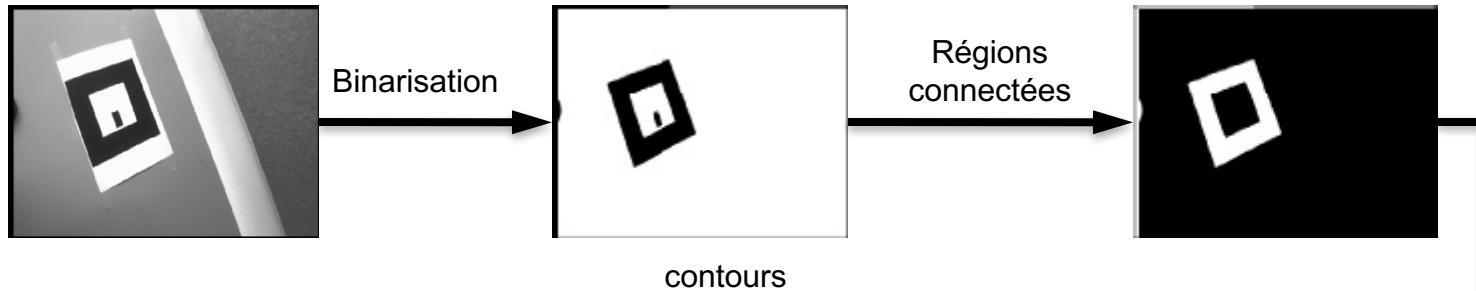
Etude de cas : Recalage / cible pour augmentation (2/5)

- Algorithme :
 - Détection de cibles 2D dans l'image
 - 2D-3D correspondances
 - Estimation de la pose = paramètres extrinsèques
 - Caméra virtuelle : paramètres intrinsèques + extrinsèques



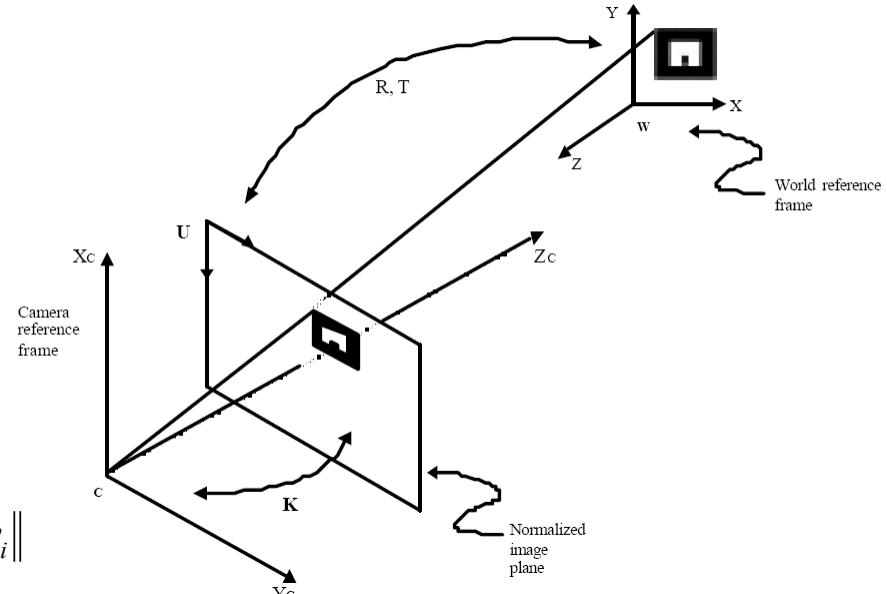
Etude de cas : Recalage / cible pour augmentation (3/5)

- Détection de cibles 2D



Etude de cas : Recalage / cible pour augmentation (4/5)

- Modèle de caméra
- Estimation de Pose
 - Estimer R & T en fonction des correspondances 2D / 3D
 - Pas d'algorithme simple !
 - Échec des méthodes linéaires
 - Minimiser $\sum_i \|Rp_i + T - v_i\|$
 - Réalité Augmentée : contrainte temps réel

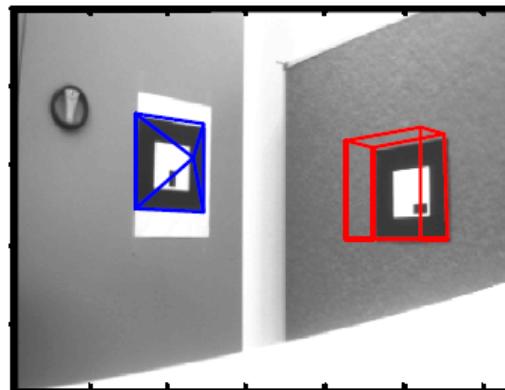


$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = (R \quad | \quad T) \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} \quad \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = (K)(R \quad | \quad T) \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} \quad K \equiv \begin{pmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

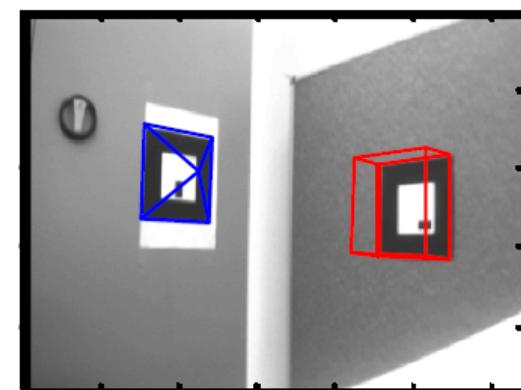
Etude de cas : Recalage / cible pour augmentation (5/5)

- Résultats

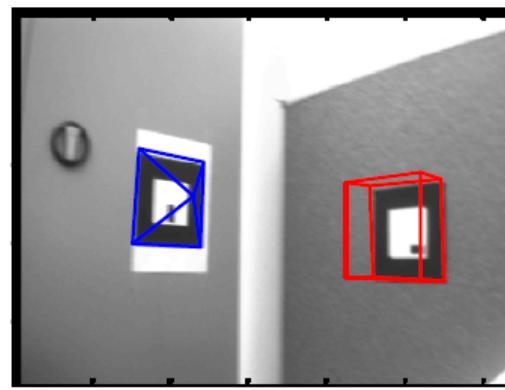
- Fiducials extraction + identification
29 ms
- Camera Pose estimation
4ms
- Augmentation time 2ms
- Total : 35ms
 \Leftrightarrow framerate \approx 28 images/s



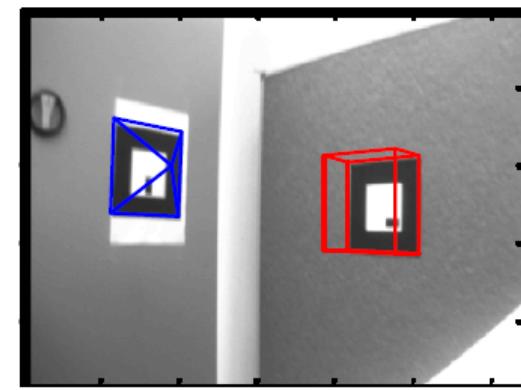
(a) frame 0



(b) frame 50



(c) frame 70



(d) frame 80

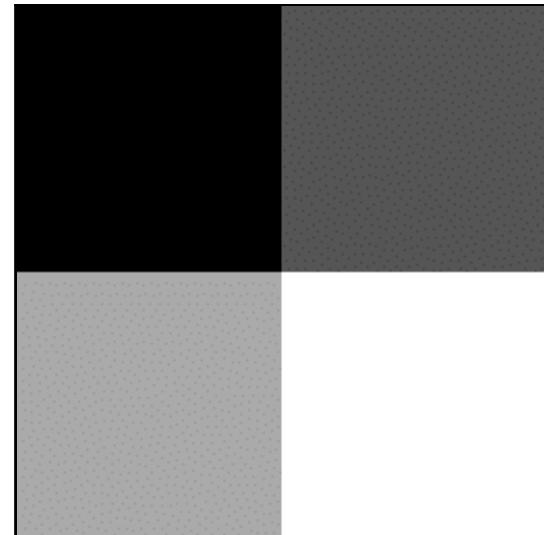
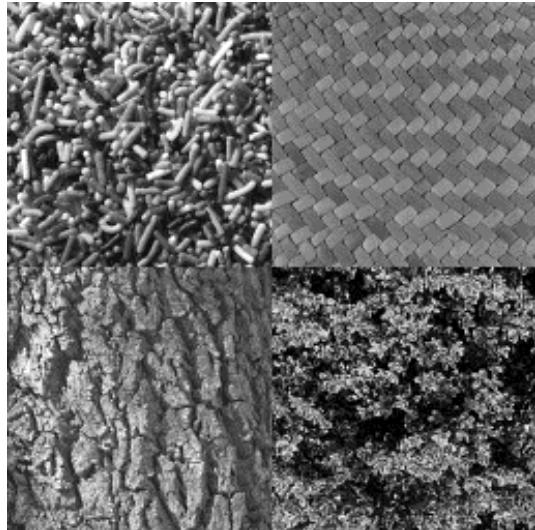


Exemples d'applications

Sylvie Lelandais / Malik Mallem

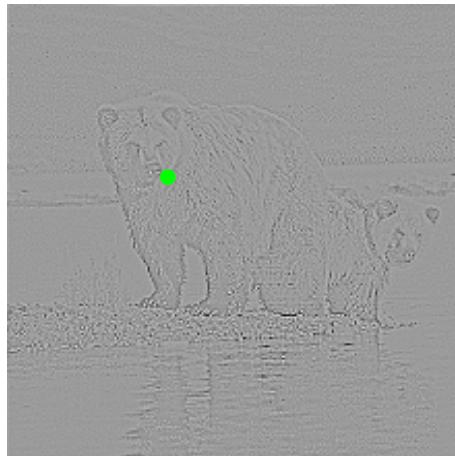
Segmentation d'images et classification de données

Problématique de la segmentation :



→ Classer les pixels en fonction de leur texture d'origine.

Exemple de décomposition



échelle : 2



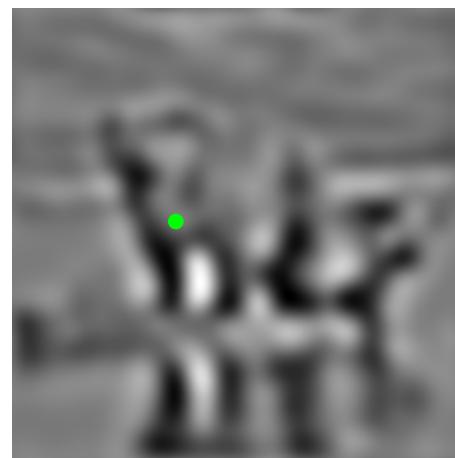
échelle : 4



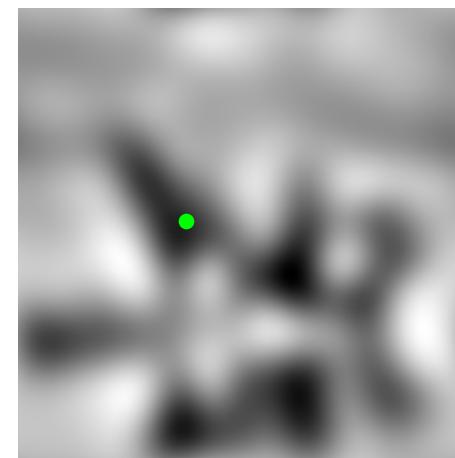
échelle : 8



échelle : 16



échelle : 32

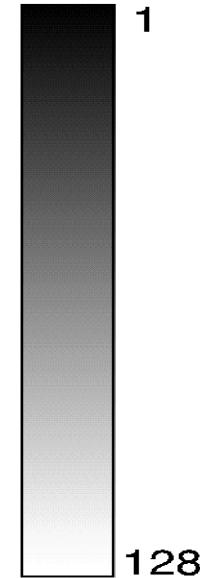
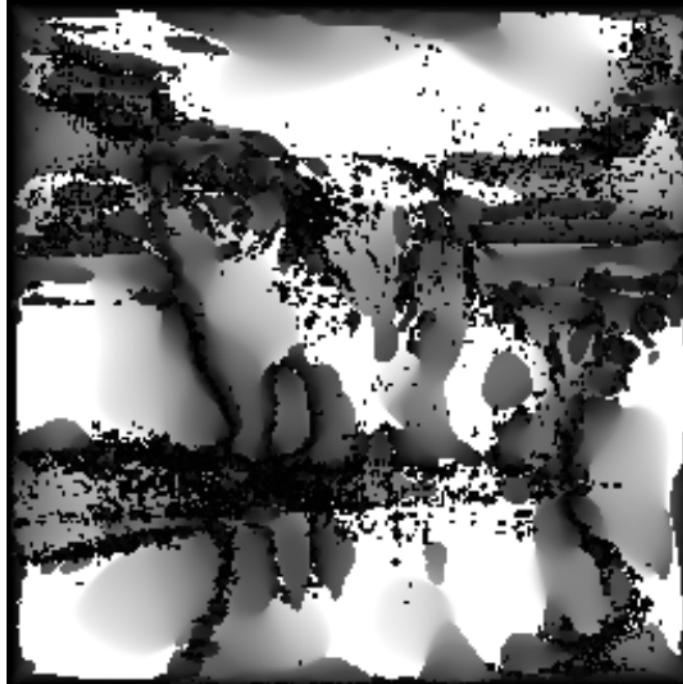
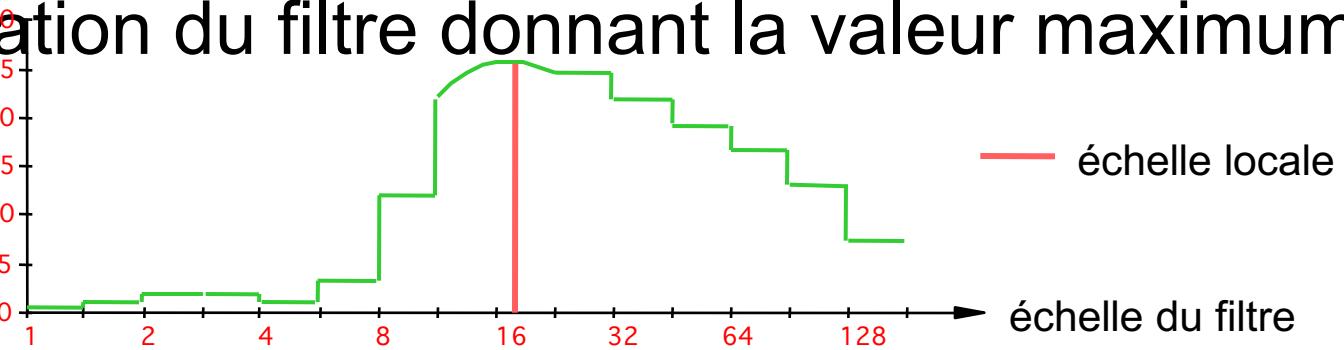


échelle : 64



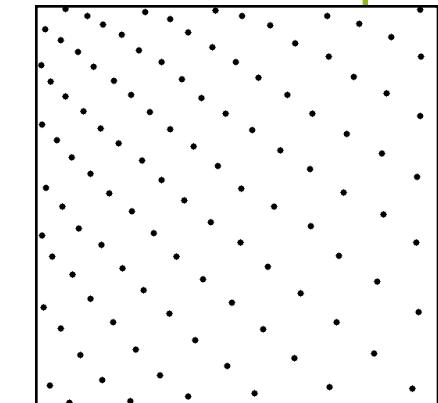
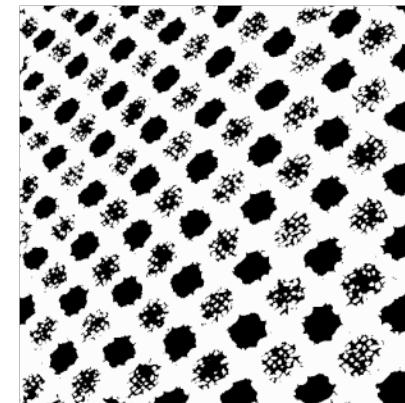
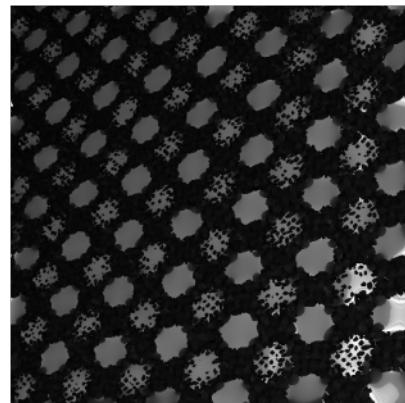
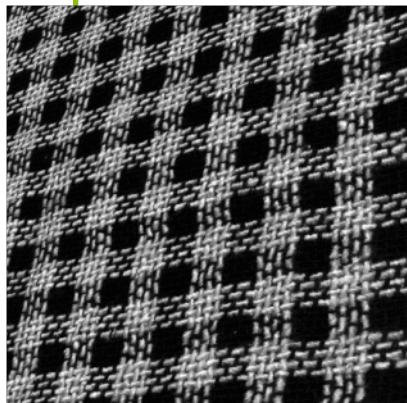
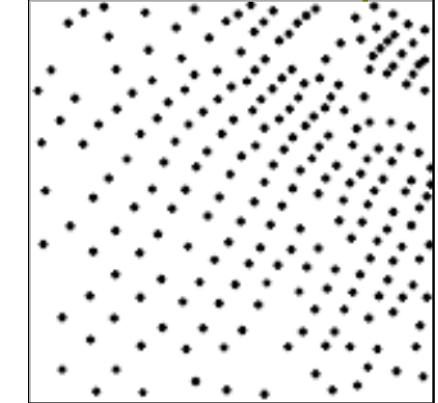
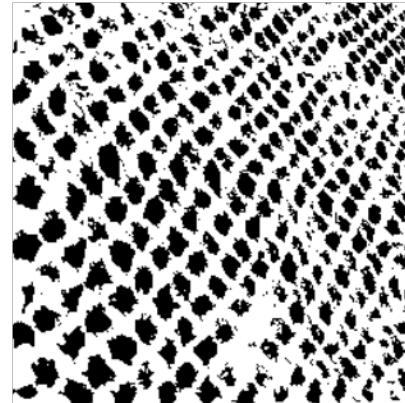
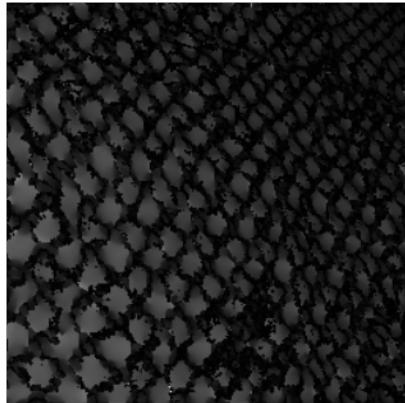
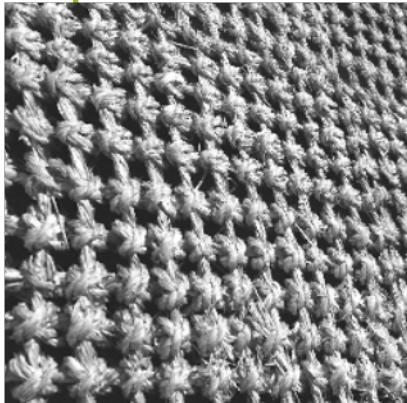
Détermination des échelles locales

Interpolation du filtre donnant la valeur maximum



échelles

Exemples d'extraction



Détection de défauts (application 2)

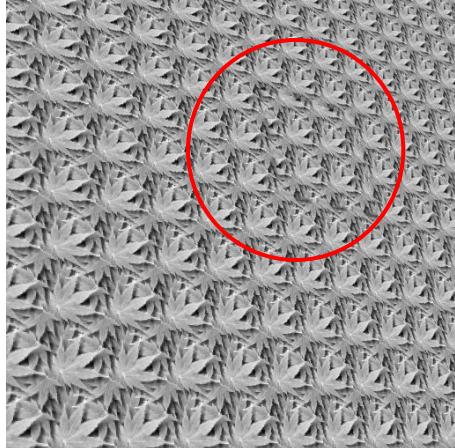
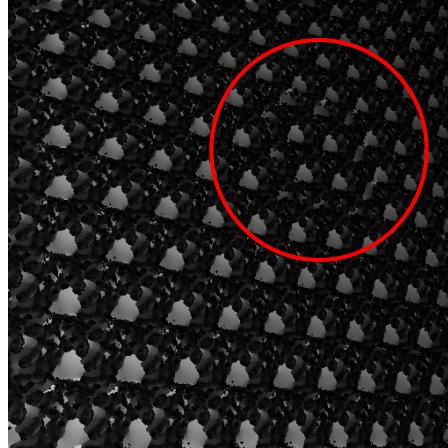
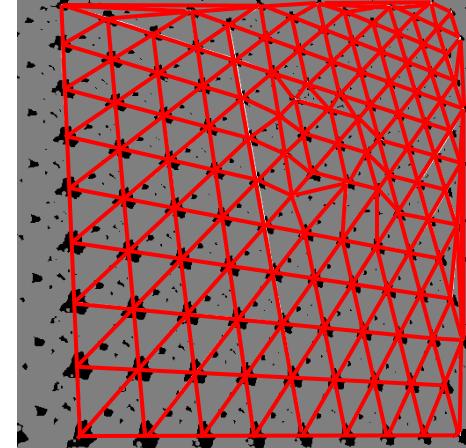


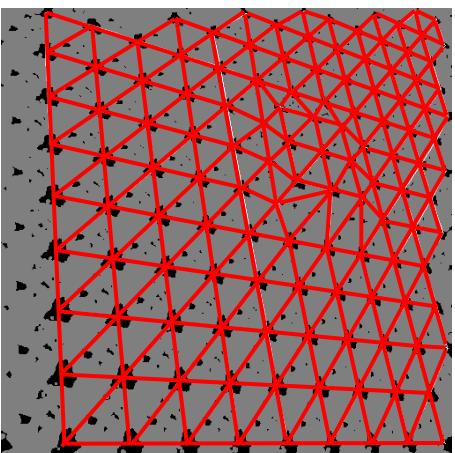
image d'origine



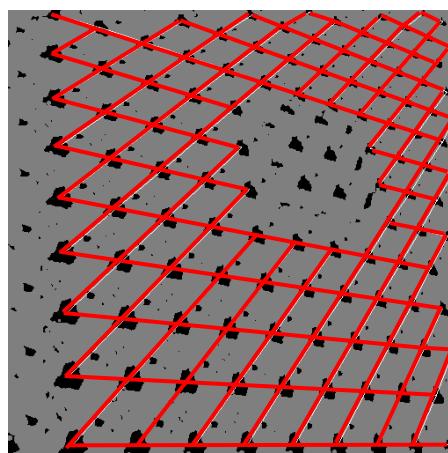
échelles locales



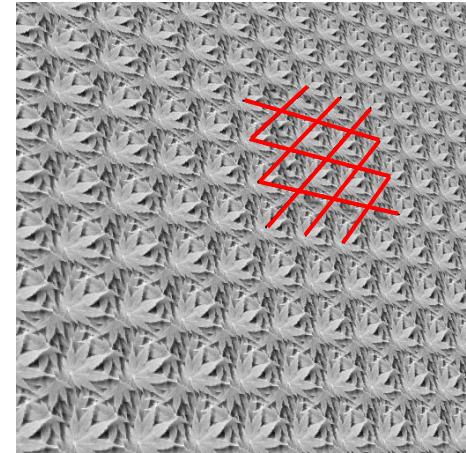
triangulation de Delaunay



triangulation simplifiée



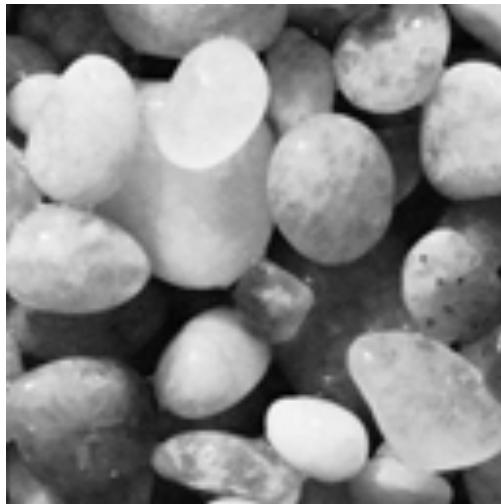
grille reconstruite



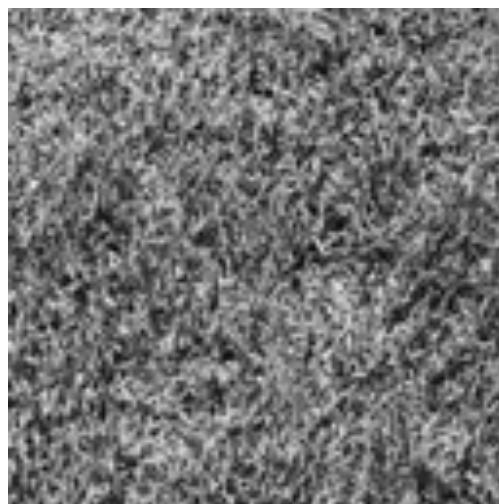
détection du défaut

A. Analyse de textures

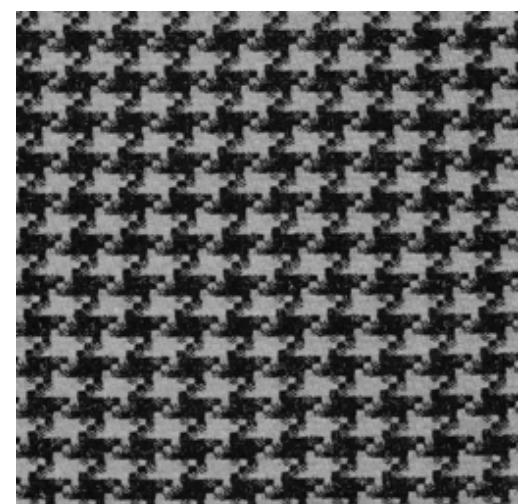
- Macrotexture : texture constituée par la répartition spatiale d'un ou plusieurs motif(s) élémentaire(s). Ces motifs sont appelés texels
- Microtexture : texture ne possédant pas de motifs élémentaires



Macrotexture naturelle



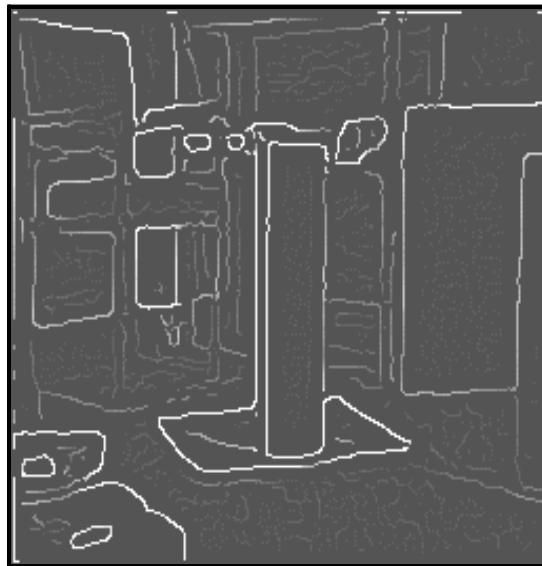
Microtexture naturelle



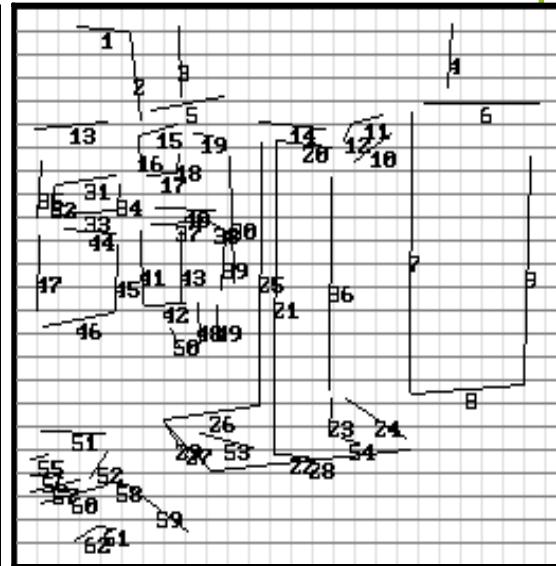
Macrotexture régulière



(a) : image originale

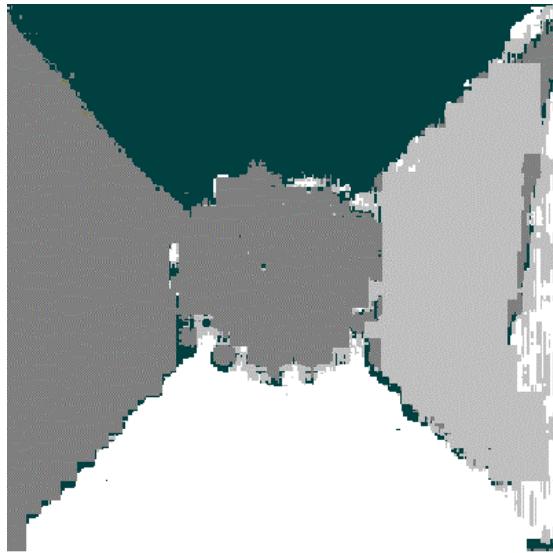
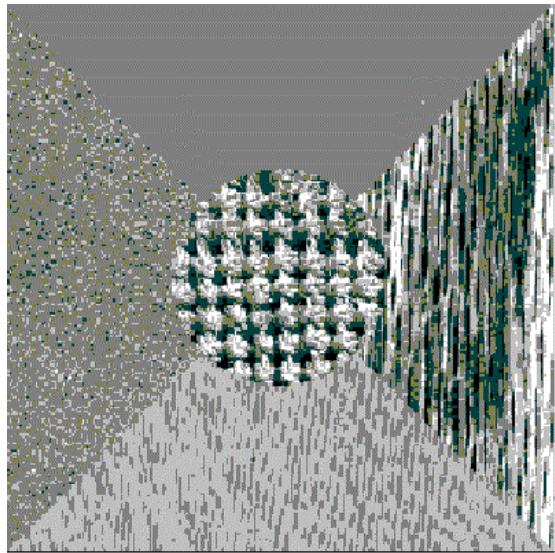


(b) : Maxima locaux



(c) : segments extraits

Segmentation d'image non texturée



(a) : image originale avec 5 textures (b) : résultat de la segmentation de 6(a)

Segmentation d'image texturée

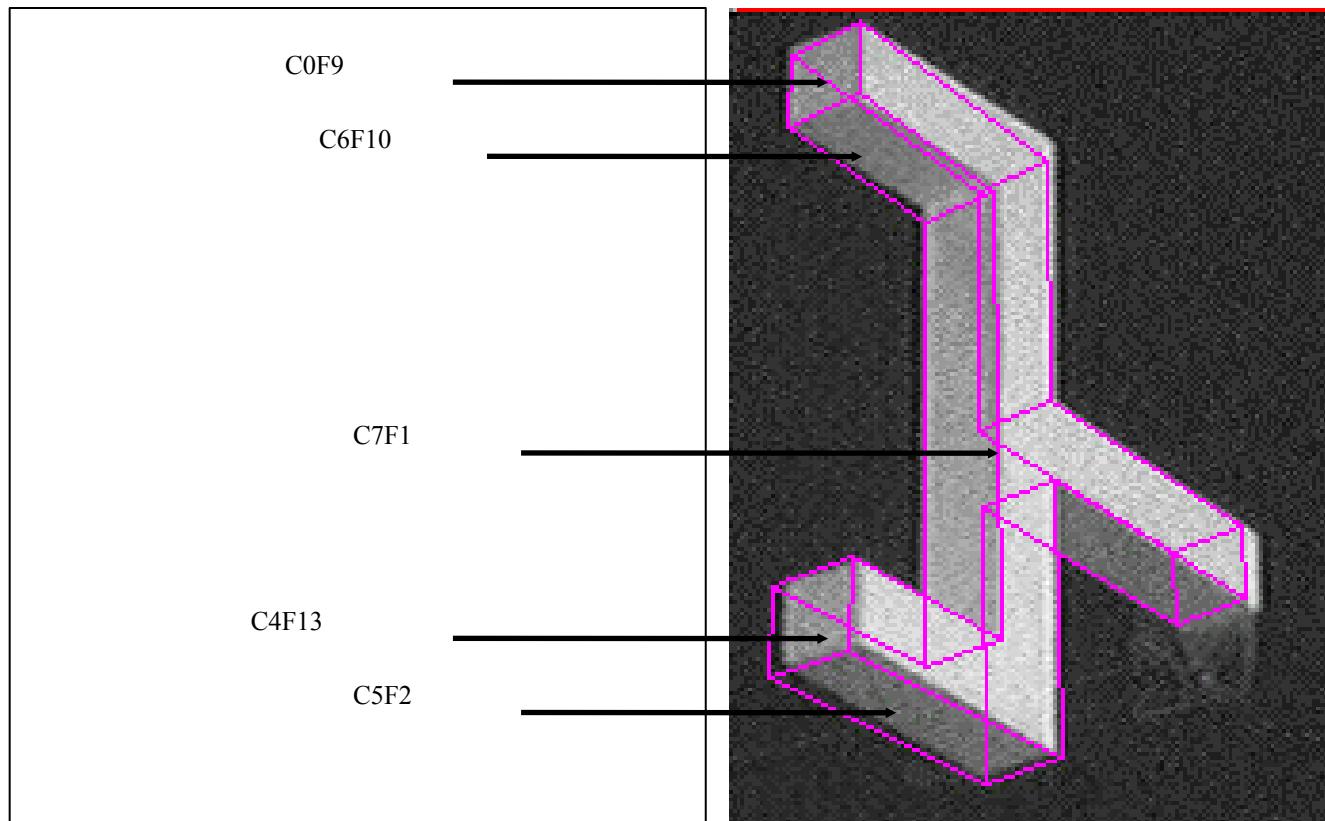
Extraction des fréquences locales par ondelettes



(a) Image originale



(b) Carte des échelles locales de (a)



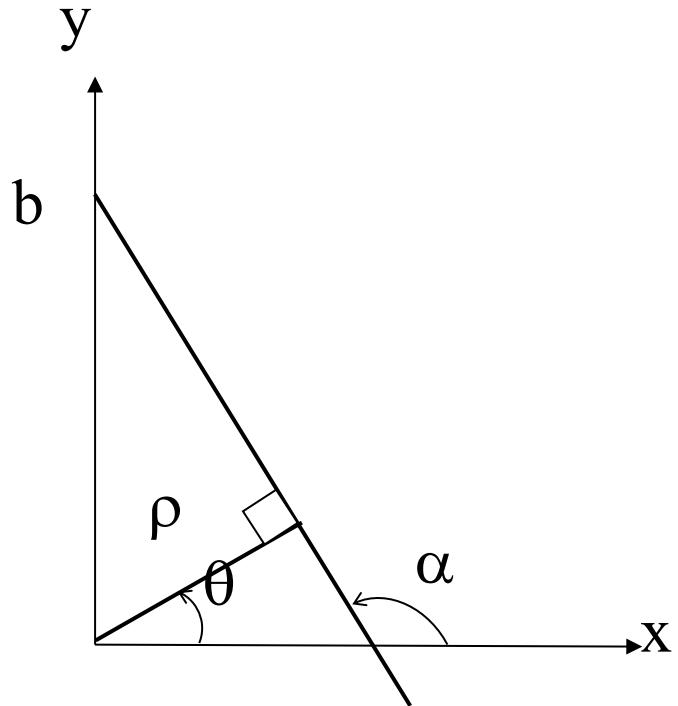
Le modèle de l'objet Pince est superposé sur son image avec erreur moyenne de 2.6 pixels, les hypothèses chaîne/face sont marquées

2. extraction de segments : Transformée de Hough : principe

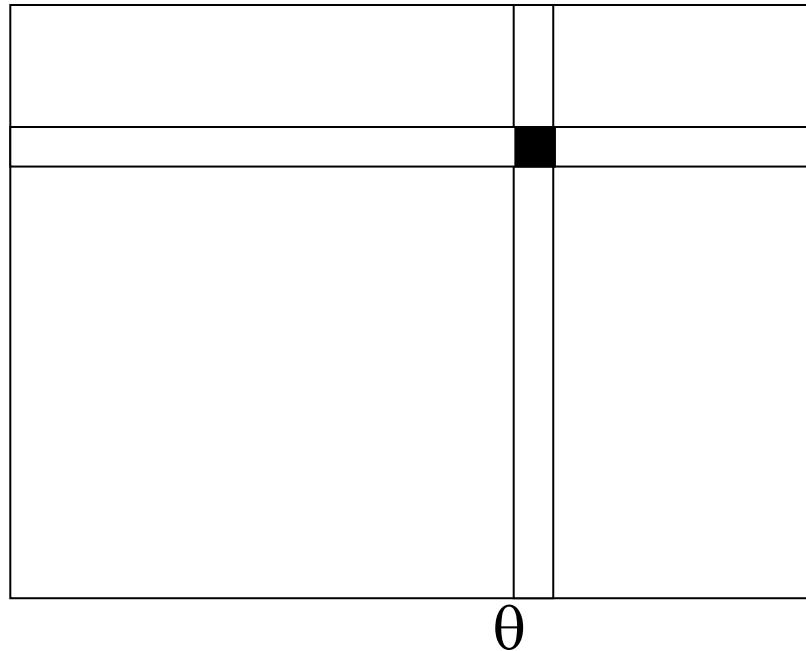
un segment



un point



ρ



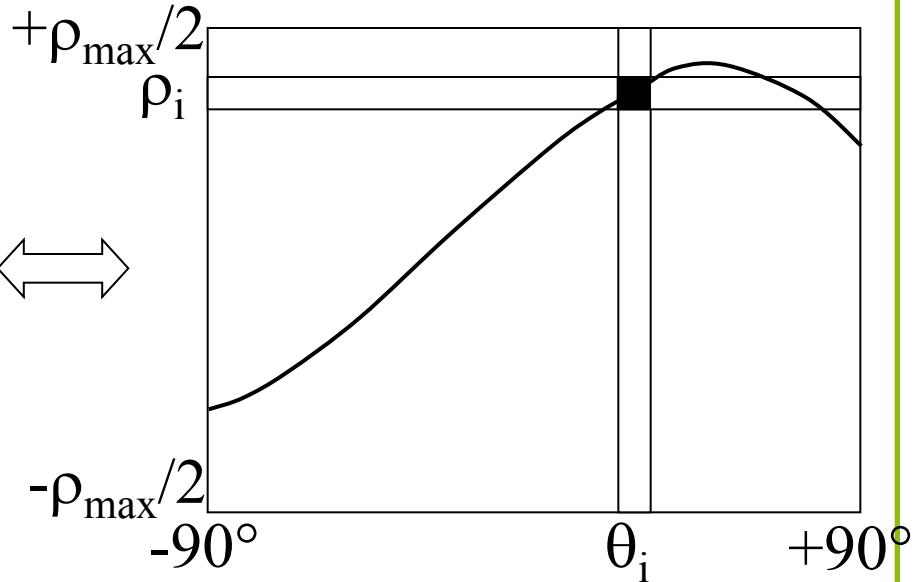
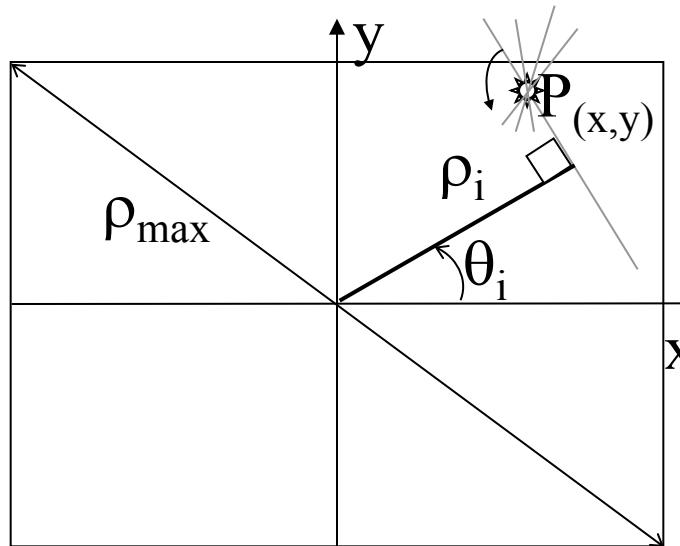
$$\rho = b \cdot \sin \theta = x \cdot \sin \theta + y \cdot \cos \theta$$

2. extraction de segments : Transformée de Hough : principe

un point



une sinusoïde



$$\rho_i = x \cdot \sin \theta_i + y \cdot \cos \theta_i$$

⌚ Mais : effet de dispersion des points sur plusieurs segments

2. extraction de segments : Transformée de Hough : principe

☺ L 'idée est d 'utiliser une tolérance sur l'angle de contour

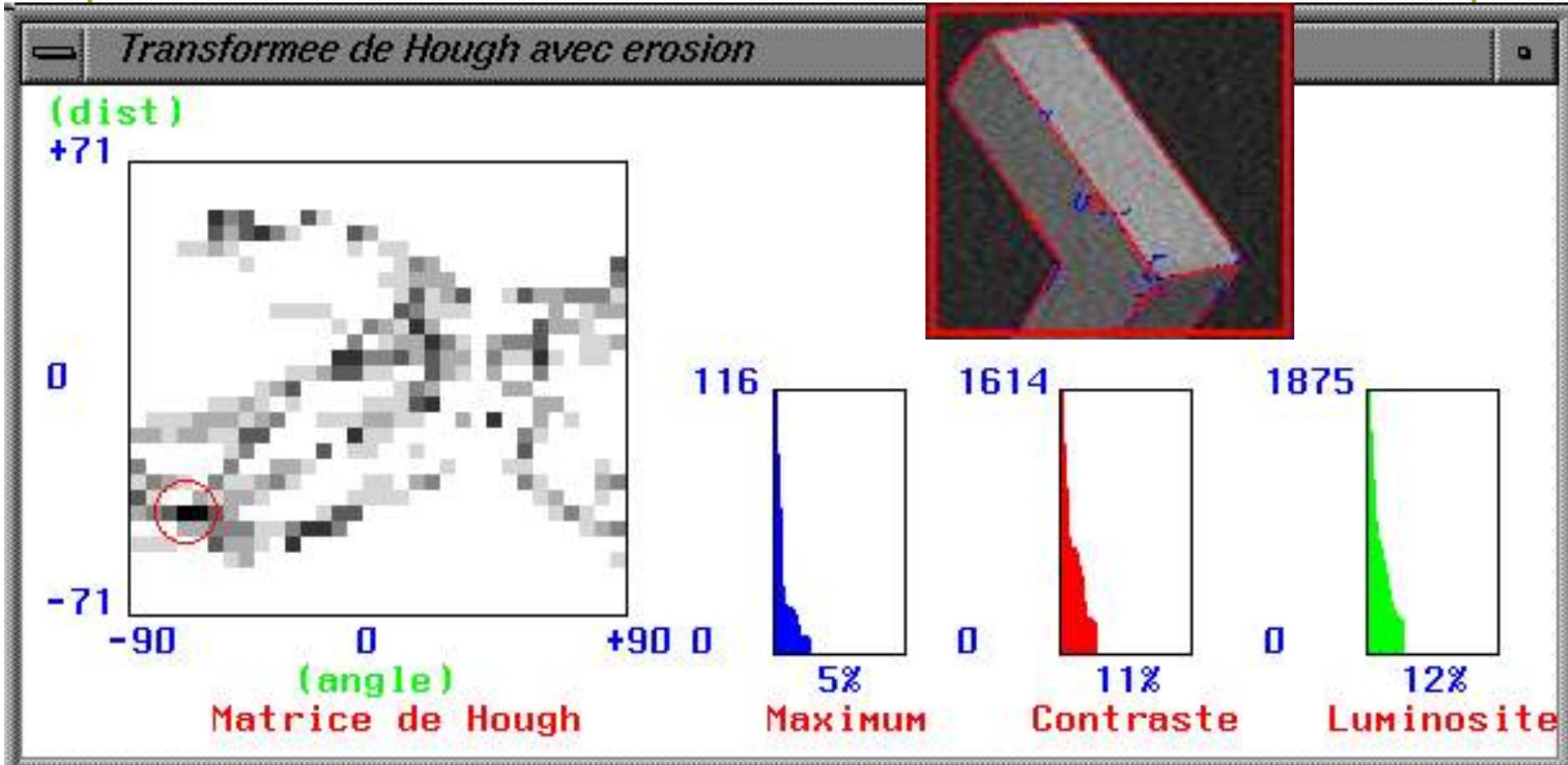
- **Algorithme:**

1. voter (nb de points) pour toutes les cellules
2. déterminer la cellule à vote maximal F le segment courant
3. retrait des points appartenant au segment courant
4. test d 'arrêt, sinon retourner à 1.

$$\text{contraste} = \sqrt{\sum_{i=1}^N (v_i - \text{moyenne})^2 / N}$$

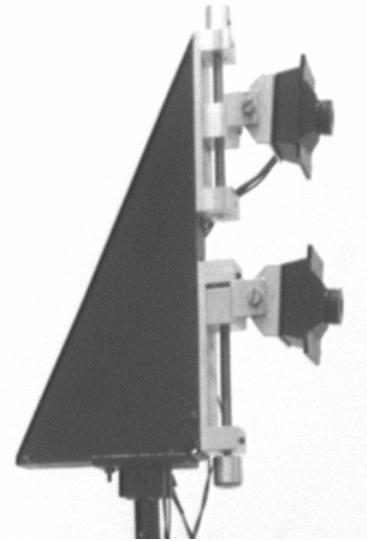
- La tolérance permet d'obtenir un meilleur résultat mais risque de confondre les segments très proches
- le critère de _____ est faiblement dépendant de la taille de l'image et le nombre de segments

2. extraction de segments : Transformée de Hough : principe



Guidage d 'une base mobile dans un environnement coloré

- Description du capteur
 - Mini-caméras
 - aucun réglage
 - caractéristiques inconnues
 - Images
 - dégradées
 - monochromatiques
 - $512 \times 480 \text{ pixels}^2$
 - Montage caméras
 - orientation: $\pm 60^\circ$
 - écart: 5 à 15 cm



Guidage d'une base mobile dans un environnement coloré



Image supérieure



Correction de
la distorsion



Image inférieure



Guidage d'une base mobile dans un environnement coloré



Image supérieure



Magnitude du gradient



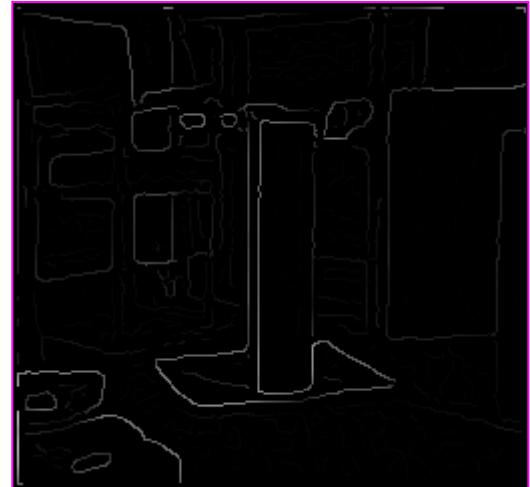
Image inférieure



Guidage d 'une base mobile dans un environnement coloré



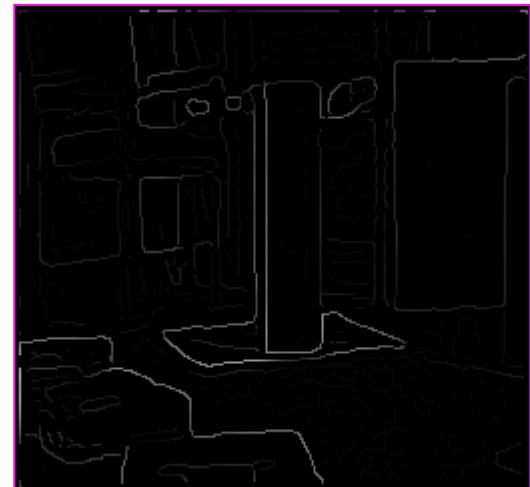
Image supérieure



Maxima locaux



Image inférieure



Guidage d 'une base mobile dans un environnement coloré



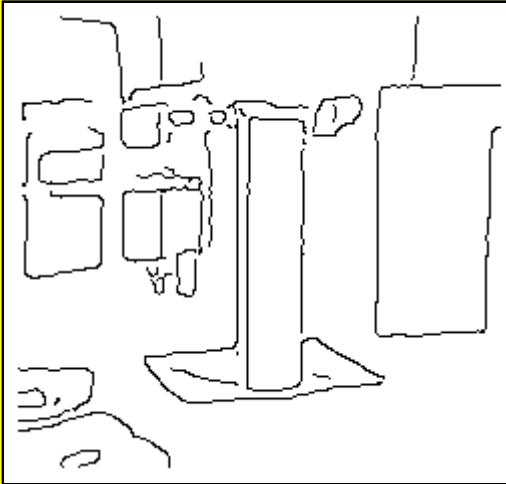
Seuil bas



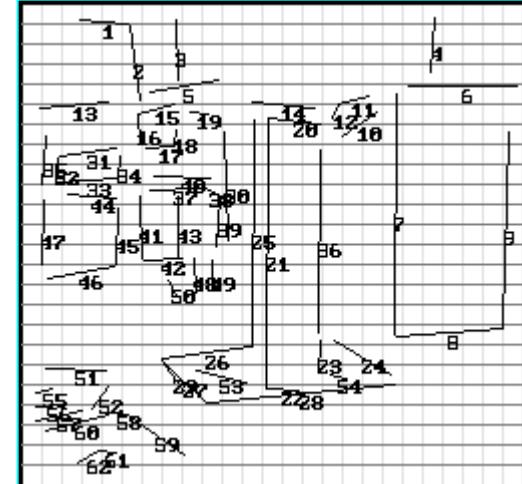
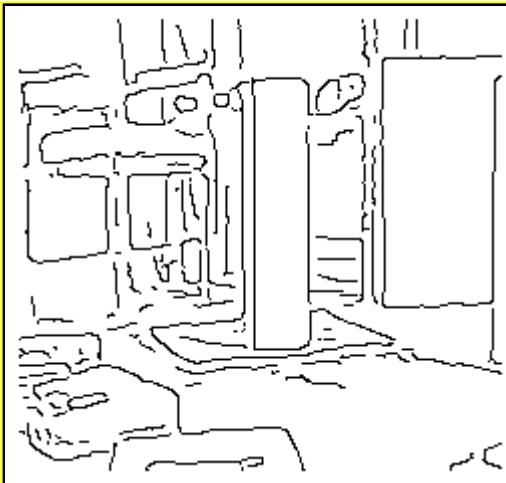
Seuil haut



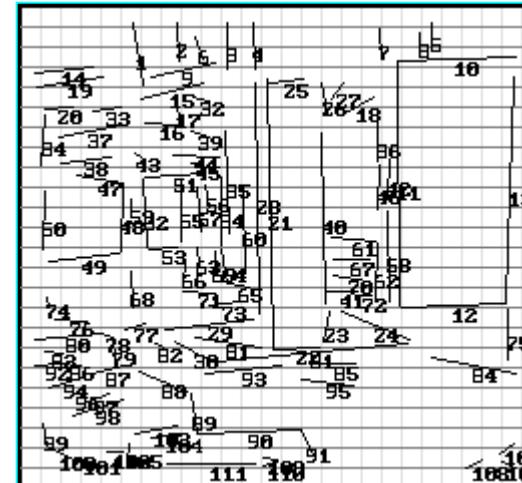
Guidage d 'une base mobile dans un environnement coloré



Pixels de contours



Segments reconstitués



Guidage d 'une base mobile dans un environnement coloré

- Nous avons utilisé 19 attributs de nature différente
- Géométriques (4)
 - coordonnées (x,y) du milieu du segment
 - longueur du segment
 - Angle β du segment
- Luminance (5)
 - moyenne de la magnitude du gradient au long du segment
 - moyenne des niveaux de gris des **régions** à droite et à gauche des segments
 - **contrastes internes** des régions à gauche et à droite des segments
- Texture (8)
 - **Définitions orientées** entre les niveaux de gris dans les régions gauche et droite des segments (quatre directions)
- Voisinage (2)
 - Liste des fenêtres intersectées par le segment
 - Liste des segments voisins