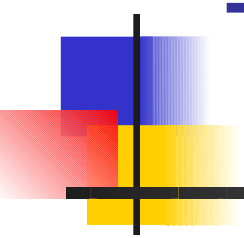


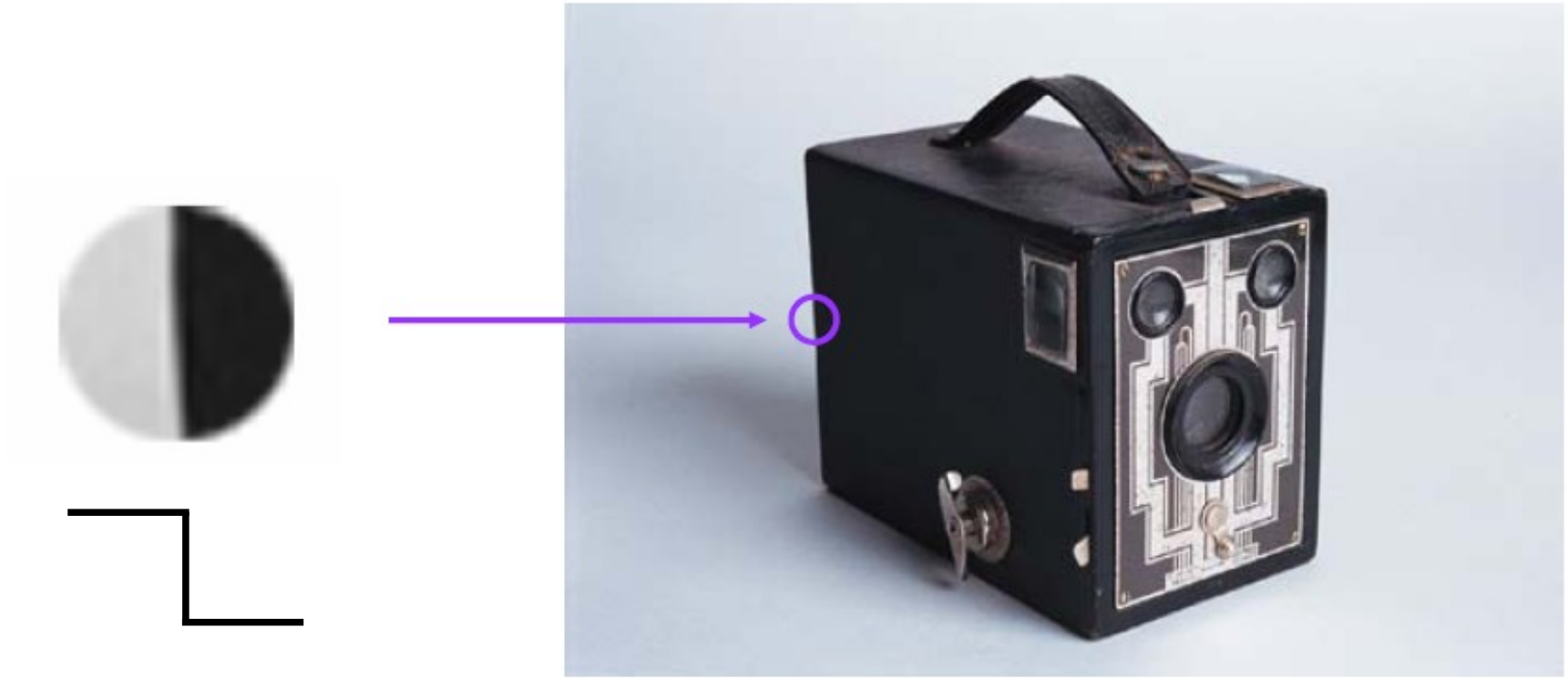
Traitement d'images

Détection de contours

Alain Boucher - IFI

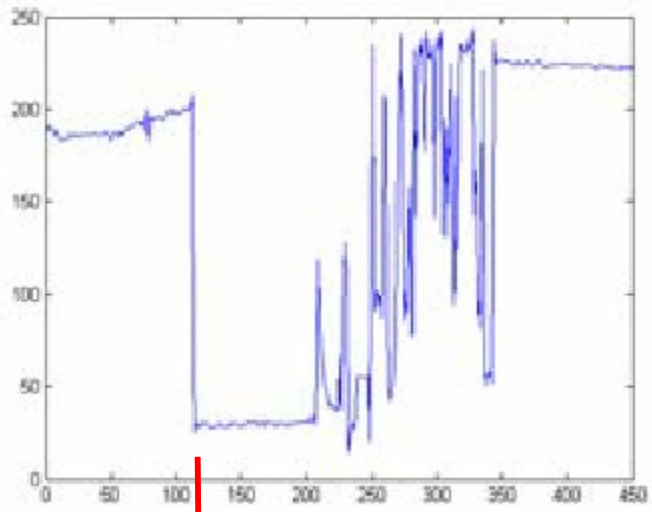
Qu'est-ce qu'un contour ?

Un contour est une variation brusque d'intensité



Source : Gluckman & Wong. Computer Vision/Scene Analysis course (CS664). Polytechnic Univ.

Qu'est-ce qu'un contour ?



Définition du contour

- Par définition, un contour est la frontière qui sépare deux objets dans une image.
  - *Une discontinuité de l'image*
- Dans notre cas, nous détecterons toutes les lignes marquant des changements d'intensité
  - *Pas seulement les contours !*
  - *Abus de langage sur la notion de contours !*

Lignes/contours dans une image

Exemples de détection des discontinuités

de profondeur

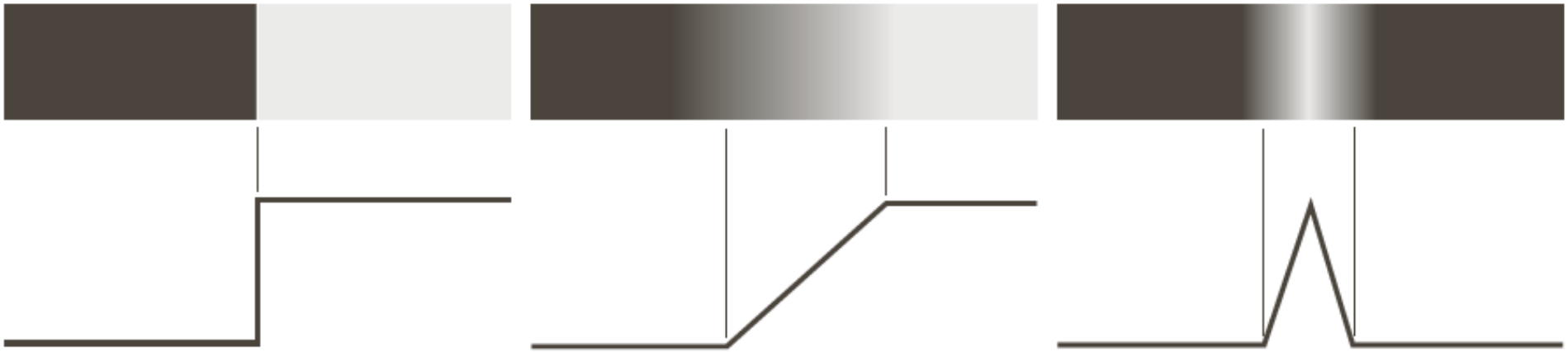
d'orientation
de surface

de réflectance

d'illumination

Source : Jacques-André Landry. Vision robotique. ETS.

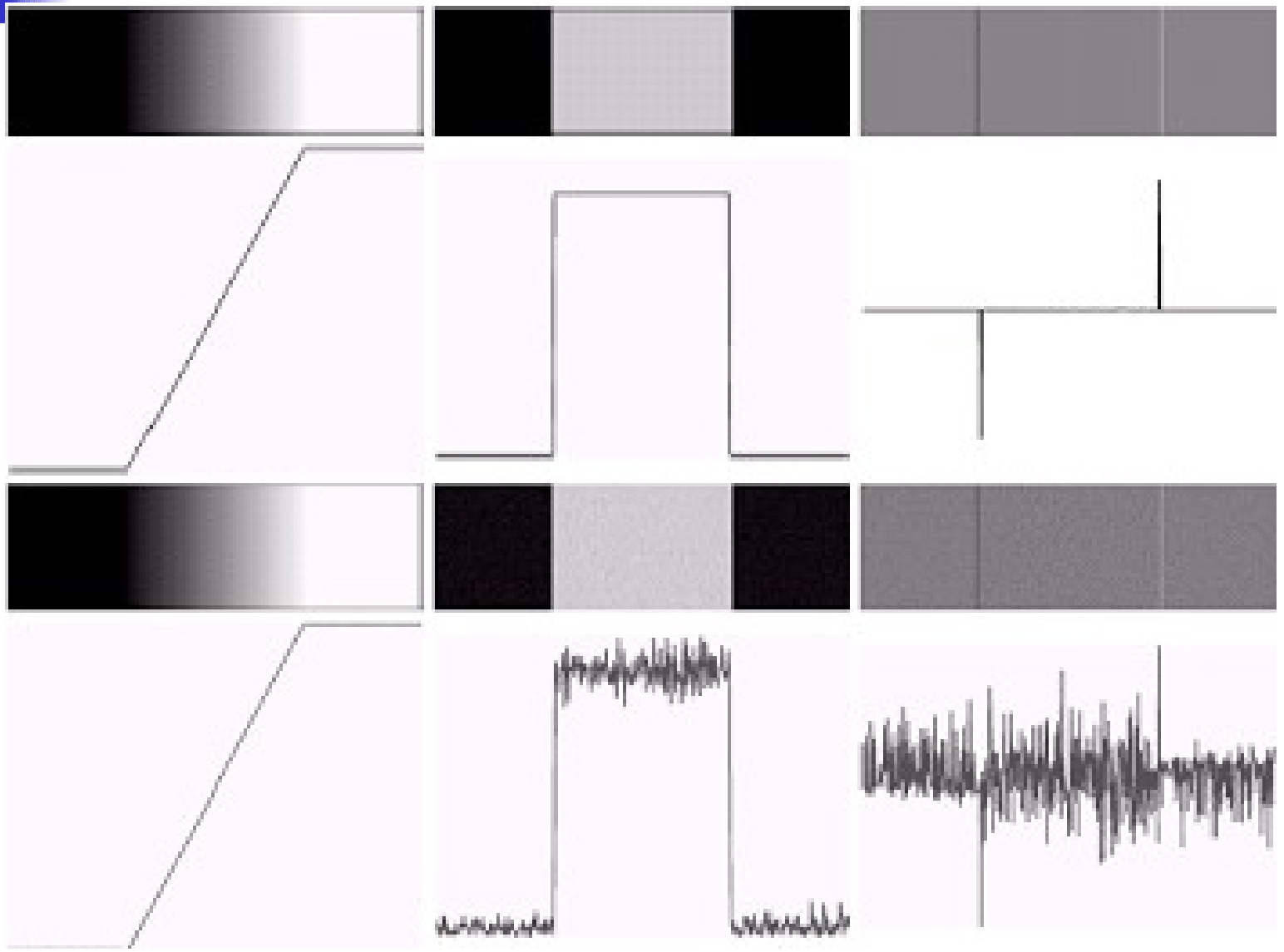
Différents types de contours



Marche d'escalier *Rampe* *Toit*

Source : Gonzalez and Woods. Digital Image Processing 3ed. Prentice-Hall, 2008.

Contour avec un peu de bruit…



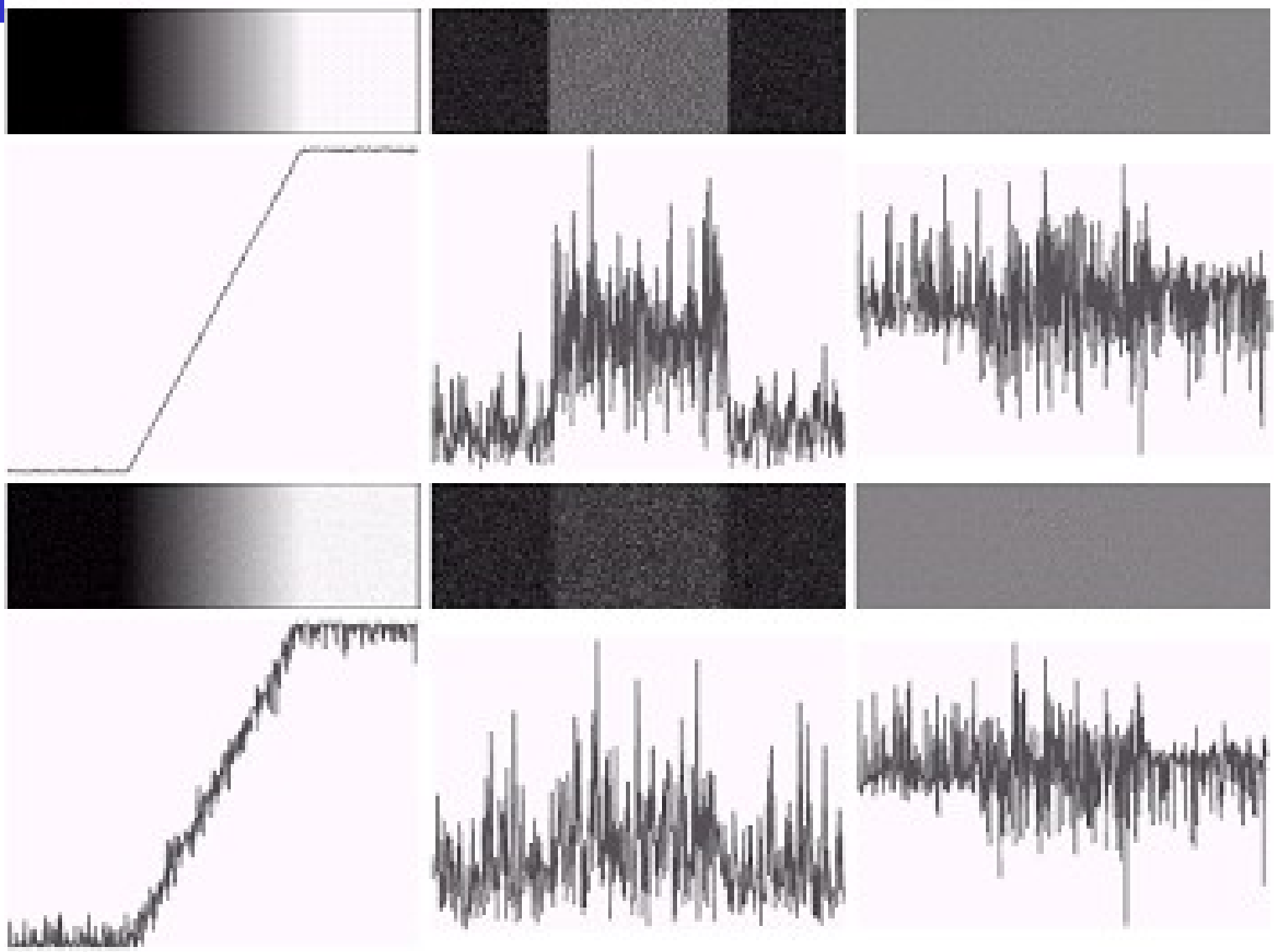
Deux types pour le filtrage spatial

- **Filtres passe-bas**
  - Atténue le bruit et les détails (basses fréquences)
    ➔ lissage

- **Filtres passe-haut**
  - Accentue les détails et les contours (hautes fréquences)
    ➔ accentuation

CVIPTools : Utilities>Filter>Specify a filter

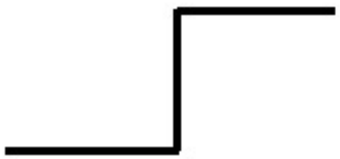
...ou beaucoup de bruit

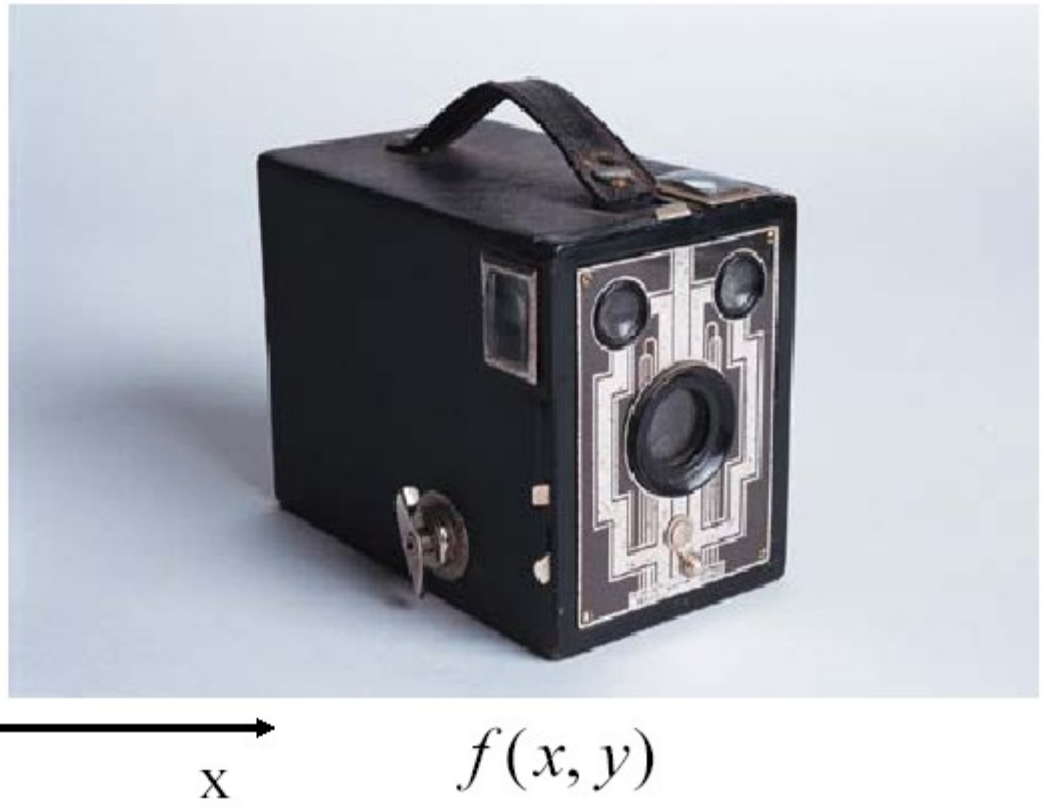


Dérivée d'une image

La première dérivée de l'image est l'opérateur de base pour mesurer les contours dans l'image

$$|\nabla f| \equiv \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2$$



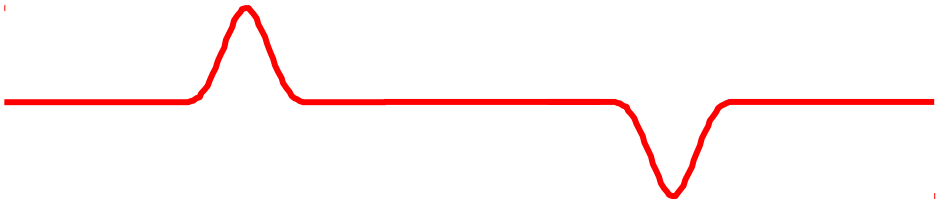


Source : Gluckman & Wong. Computer Vision/Scene Analysis course (CS664). Polytechnic Univ..

Dérivée d'une image et contours

Image 1D f(x)

1ère dérivée f'(x)



|f'(x)|

Pixels contours:

|f'(x)| > Seuil

Dérivée discrète

On utilise la première dérivée de l'image pour les contours :

$$\frac{\Delta I}{\Delta x} = \frac{I(x+\Delta x) - I(x)}{\Delta x}$$

Approximation simple de la dérivée discrète :

-1	1

-1
1

ou encore :

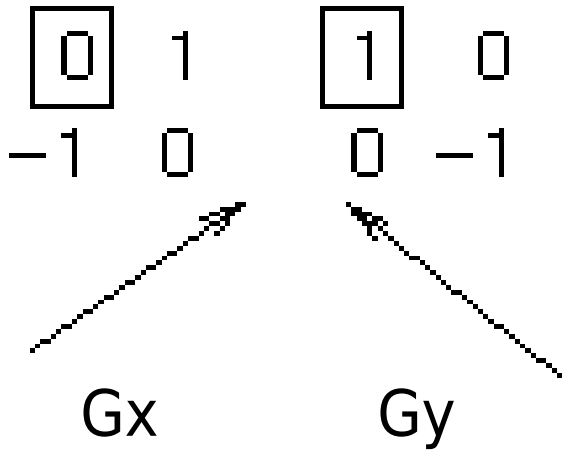
-1	0	1

-1
0
1

Filtre de Roberts

- Roberts (1965) fournit une première approximation de la première dérivée d'une image discrète

- Le calcul se fait avec 2 masques de convolution pour les 2 directions de la dérivée



Filtres pour la détection de contours

Plusieurs autres filtres existent pour la détection des contours dans l'image

On fait lissage de l'image + dérivée de l'image (sauf Roberts)

Il existe beaucoup d'autres filtres pour détecter les contours

Roberts:

1	0
0	-1

0	1
-1	0

Prewitt:

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

Sobel:

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

CVIPTools : Analysis>Edge/Line Detection

Lissage + dérivée de l'image

Filtre de Prewitt : Moyenneur + Dérivée

$$\begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 & 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} * \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$$

Filtre de Sobel : Gaussienne + Dérivée

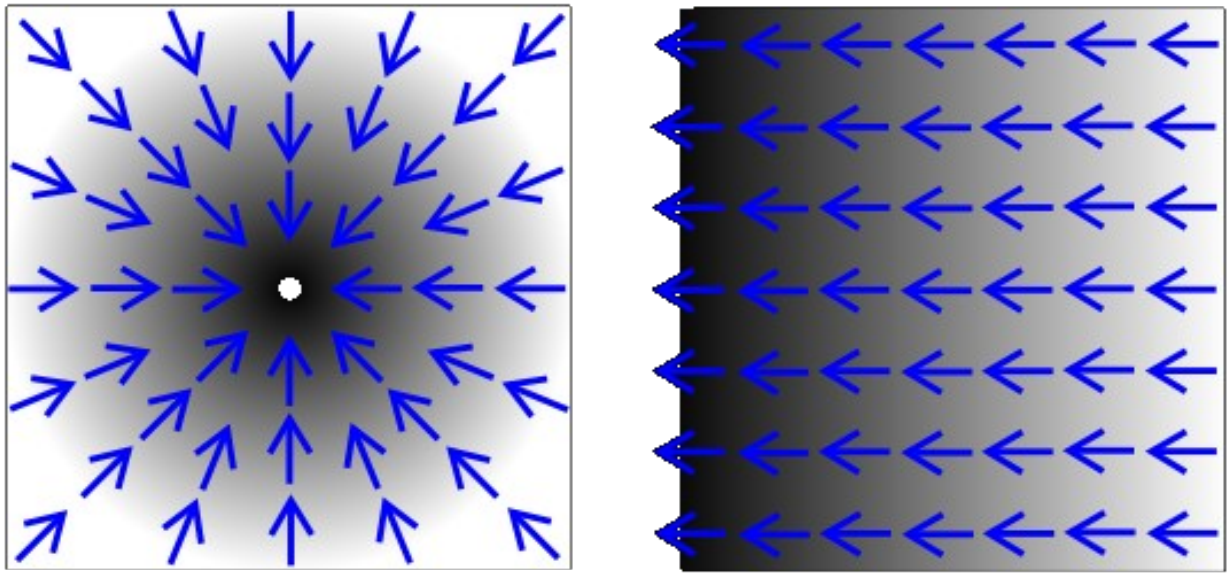
$$\begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} * \begin{pmatrix} -1 & 0 & 1 \end{pmatrix} \qquad \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} * \begin{pmatrix} 1 & 2 & 1 \end{pmatrix}$$

⇒ Détection des contours moins sensible au bruit

Gradient de l'image

- En 2 dimensions, nous parlons de gradient de l'image
  - *dérivée en X + dérivée en Y*

→ vecteur avec une norme et une direction



Les lignes bleues représentent le gradient de couleur du plus clair vers le plus foncé

Source : Gradient. Wikipédia (fr).

Gradient : norme et direction

Norme : Intensité du gradient en chaque pixel

(mesure la plus utilisée)

$$|G| = \sqrt{Gx^2 + Gy^2} \approx |Gx| + |Gy|$$

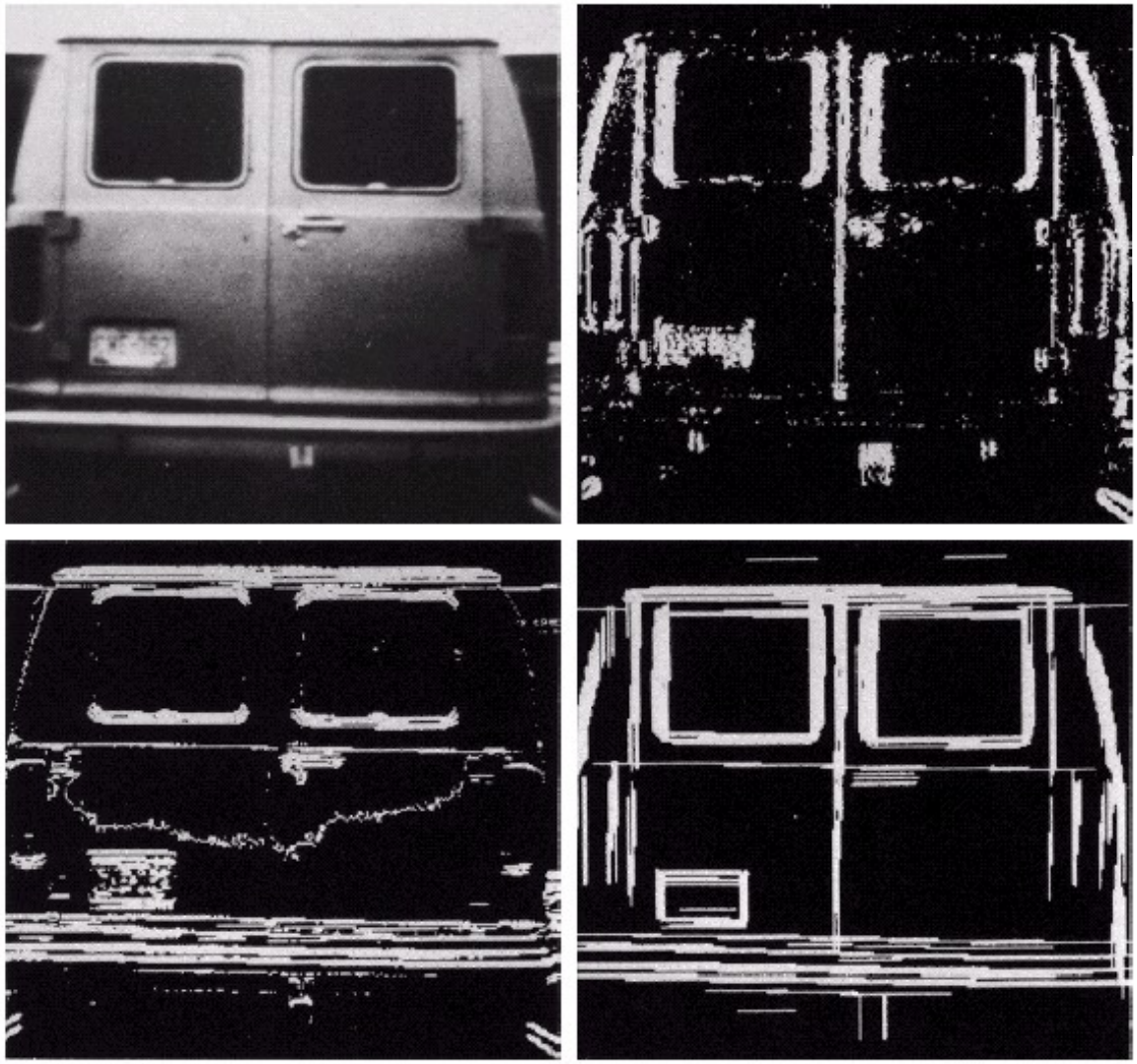
Direction : Direction du gradient le plus fort en chaque pixel

$$\theta = \arctan\left(Gy / Gx\right)$$

Exemple de détection de contours

a b
c d

FIGURE 10.16
(a) Input image.
(b) G_y component of the gradient.
(c) G_x component of the gradient.
(d) Result of edge linking. (Courtesy of Perceptics Corporation.)



$\frac{\partial f}{\partial x}$

Contours verticaux

$\frac{\partial f}{\partial y}$

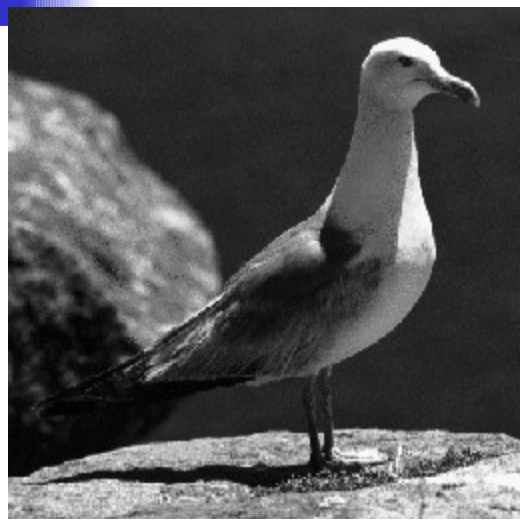
Contours horizontaux

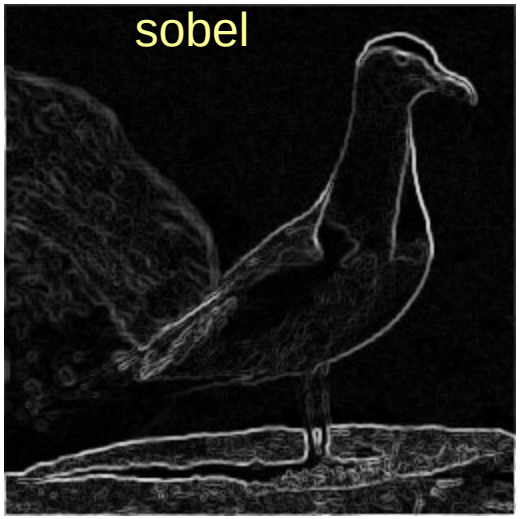
$\frac{\partial f}{\partial y} + \frac{\partial f}{\partial x}$

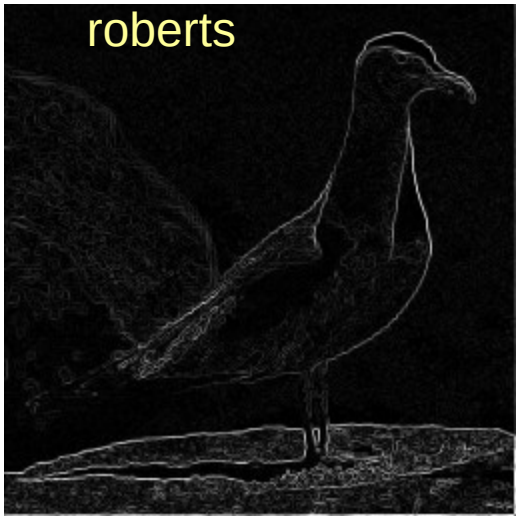
Norme

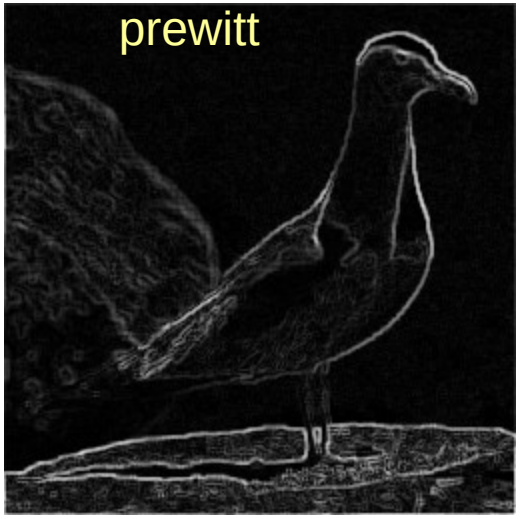
Source : Gonzalez and Woods. Digital Image Processing. Prentice-Hall, 2002.

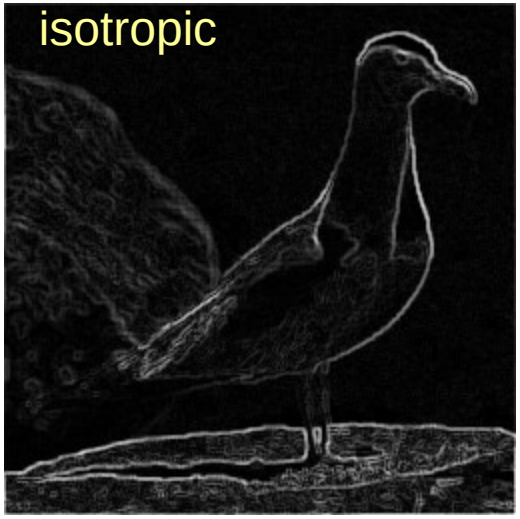
Exemples de détections de contours







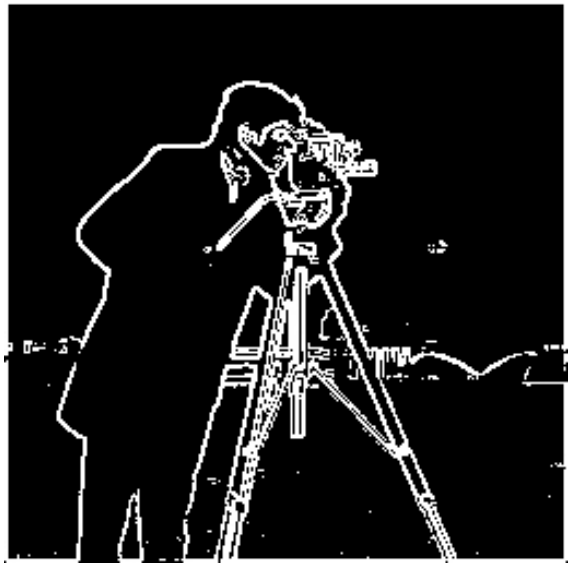




Seuillage du gradient

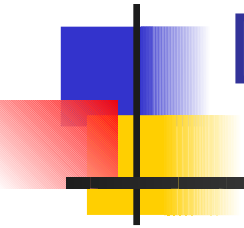
Détection
avec Sobel
sans seuillage

Seuillage
avec S=25



Seuillage
avec S=60

Source : Caroline Rougier. Traitement d'images (IFT2730). Univ. de Montréal.



Deuxième dérivée de l’image

Laplacien

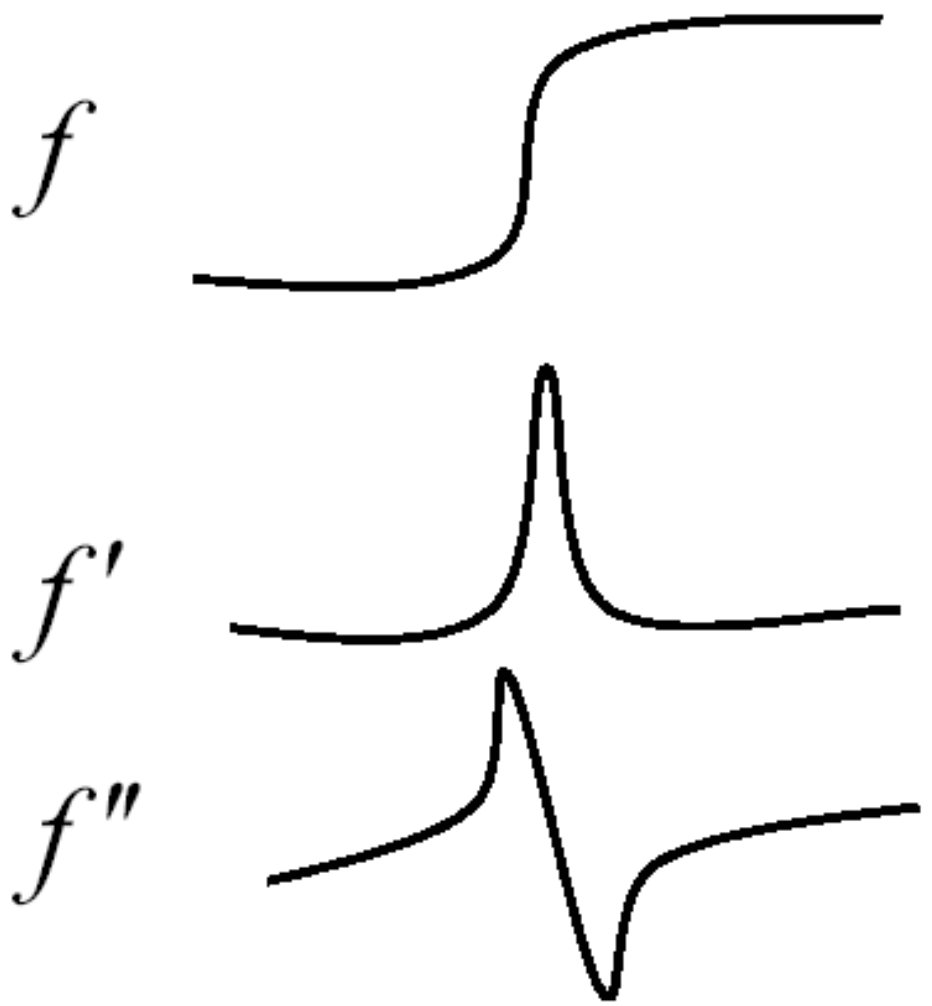
Deuxième dérivée de l'image

- Une autre approche pour trouver les contours de l'image est d'utiliser la seconde dérivée de l'image

- Pour cela, on utilise le Laplacien comme opérateur

$$\nabla^2 I = \frac{\partial I}{\partial x^2} + \frac{\partial I}{\partial y^2}$$

Dérivées de l'image

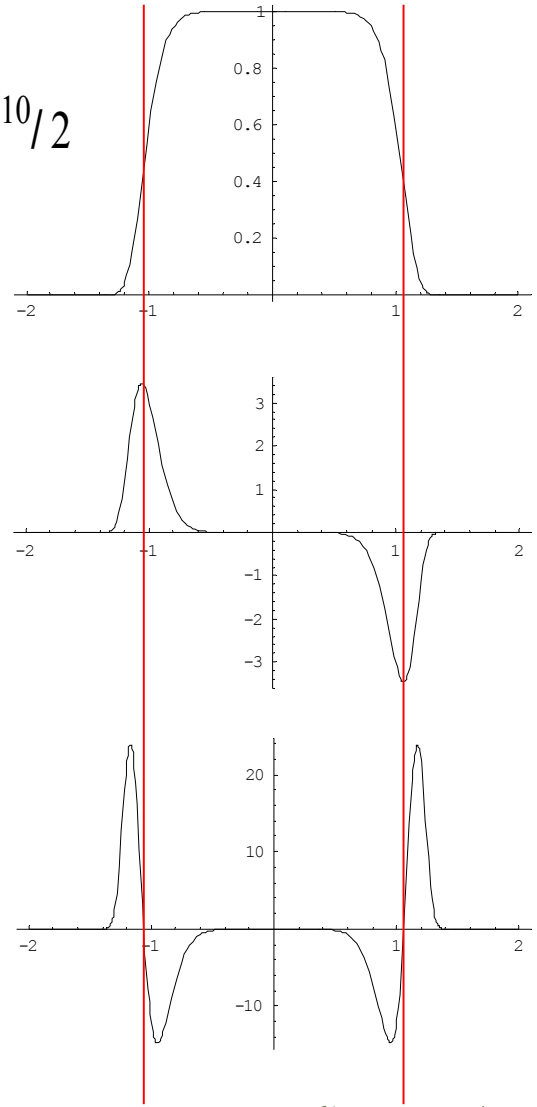


Les contours correspondent :

- Aux maxima de la première dérivée
- Aux passages par zéros de la deuxième dérivée

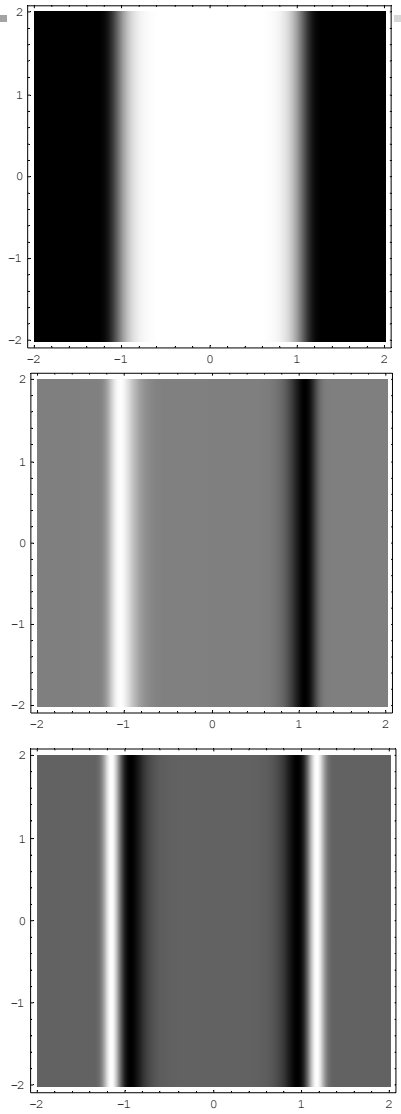
Dérivées de l'image

$$f(x,y)=e^{-x^{10}/2}$$



$$\frac{\partial f}{\partial x}$$

$$\frac{\partial^2 f}{\partial x^2}$$



Image

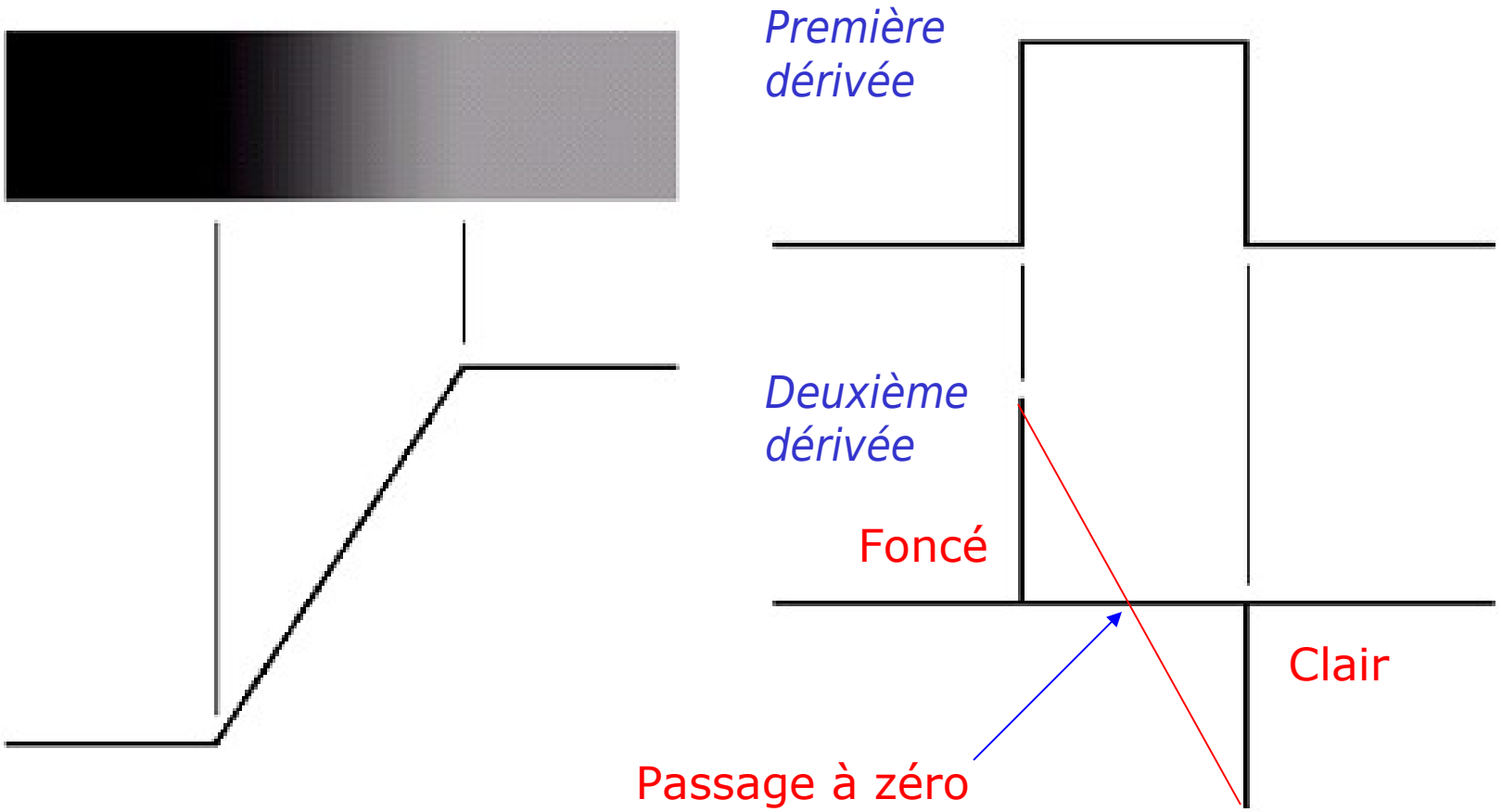
Première dérivée

Deuxième dérivée

Source : Caroline Rougier. Traitement d'images (IFT2730). Univ. de Montréal.

Exemple avec un contour « rampe »

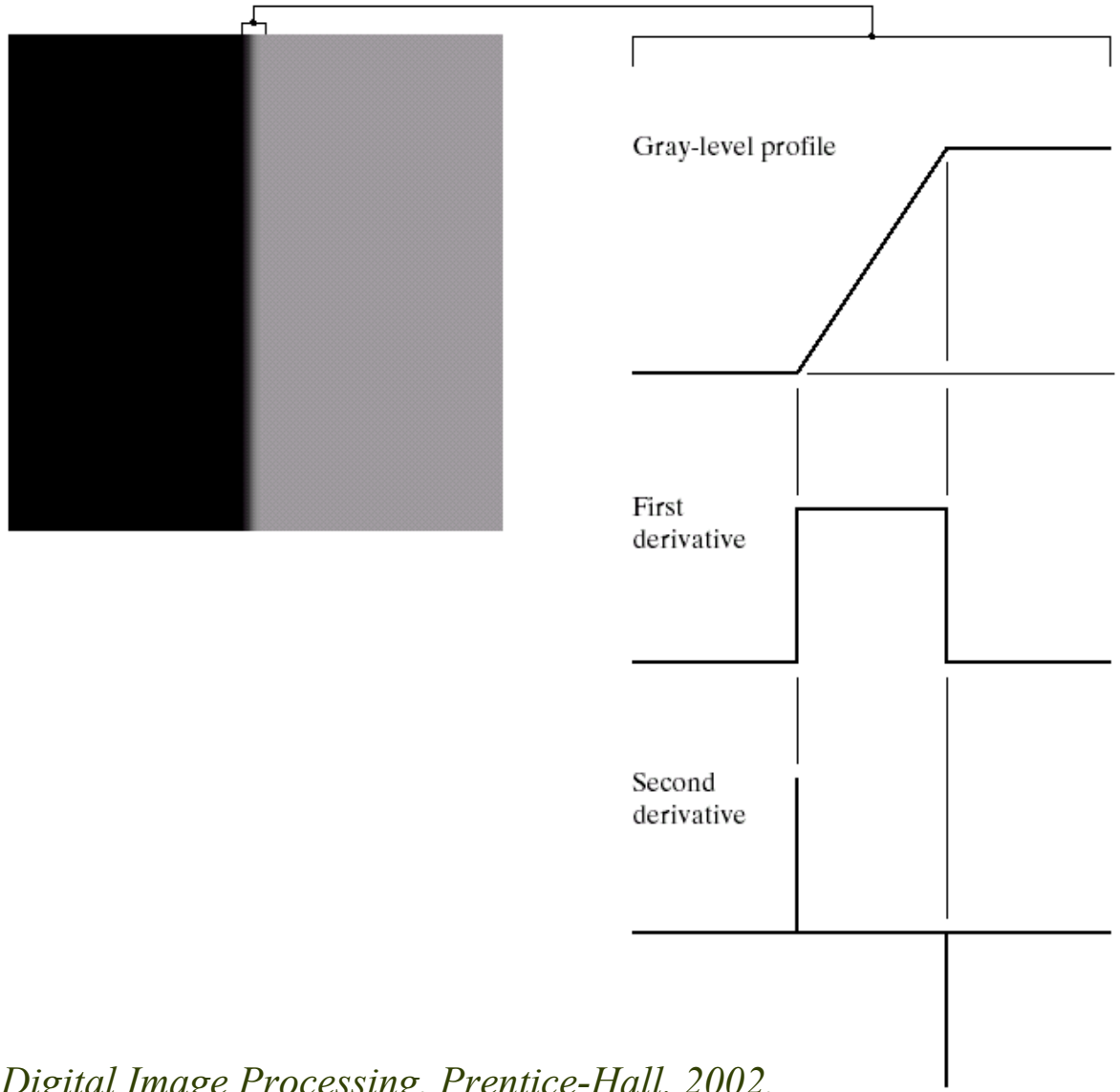
Détection de la frontière



Dérivées de l'image

a b

FIGURE 10.6
(a) Two regions separated by a vertical edge. (b) Detail near the edge, showing a gray-level profile, and the first and second derivatives of the profile.



Source : Gonzalez and Woods. Digital Image Processing. Prentice-Hall, 2002.

Laplacien par convolution

- Plusieurs approximations discrètes du Laplacien existent

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{ou} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

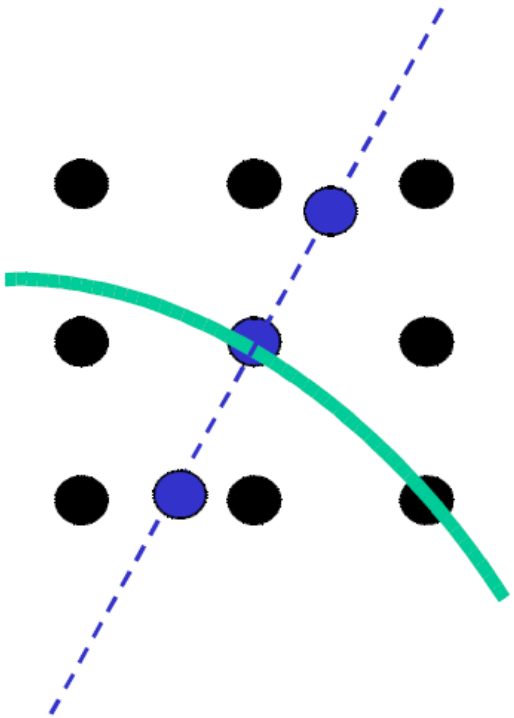
- Une seule matrice de convolution
- Symétrique en rotation

CVIPTools : Analysis>Edge/Line Detection>Laplacian

Canny filter: steps (2)

4 – Non-maxima suppression

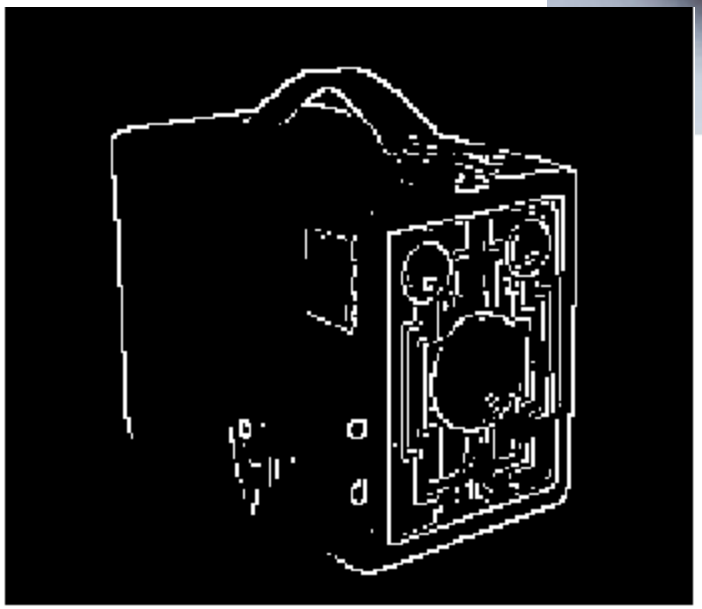
- If the gradient magnitude of a pixel (x,y) is inferior to the one of its 2 neighbors along the gradient direction
  - then set this magnitude for (x,y) to zero



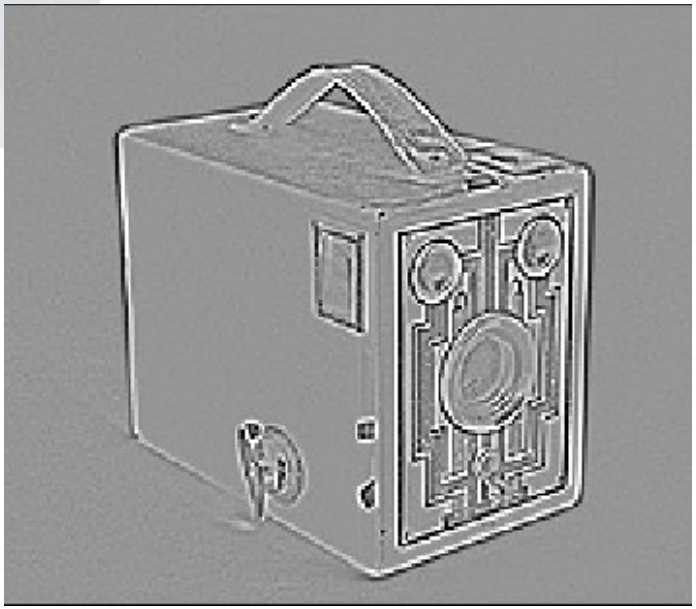
Source : Thomas Boudier, UPMC, www.snv.jussieu.fr/~wboudier/ens/cours_inb2/06_Segmentation.pdf

Comparaison Gradient / Laplacien

Gradient



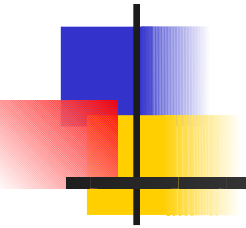
Laplacien



Quel filtre choisir pour les contours ?

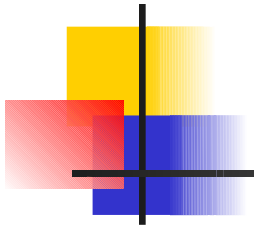
- Aucun opérateur n'est parfait pour détecter les contours
- En pratique, on obtient des contours incomplets
  - *il y a des pixels superflus*
  - *il y a des manques*
  - *il y a des erreurs de position et d'orientation des pixels contours*

- Chacun semble avoir sa préférence pour une méthode ou une autre
- Un opérateur de détection de contour n'est qu'une première étape dans la chaîne de segmentation



Filtrage optimal

Filtre de Canny



Filtrage optimal : Canny

- Filtre optimal pour la détection des contours
  - *Filtre en plusieurs étapes (pas seulement une convolution)*
- Etant donnés
  - un modèle de contour (marche)
  - un modèle de bruit (blanc gaussien)
- Caractériser les performances en termes de :
  - **détection** (surtout pour les contours faibles)
  - **localisation** (contour détecté proche du contour réel)
  - **réponse unique** (un contour = une seule réponse)

Filtre de Canny : étapes (1)

1 – Appliquer un filtre Gaussien sur l'image

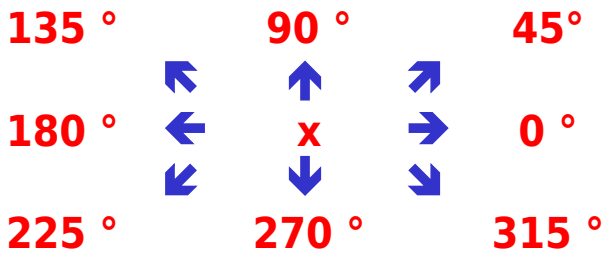
- Filtre passe-bas pour enlever le bruit

2 – Calculer l'intensité du gradient dans l'image

- Filtre de Sobel en X et Y
- Calcul de la norme $|G| = |Gx| + |Gy|$

3 – Calculer les directions du gradient dans l'image

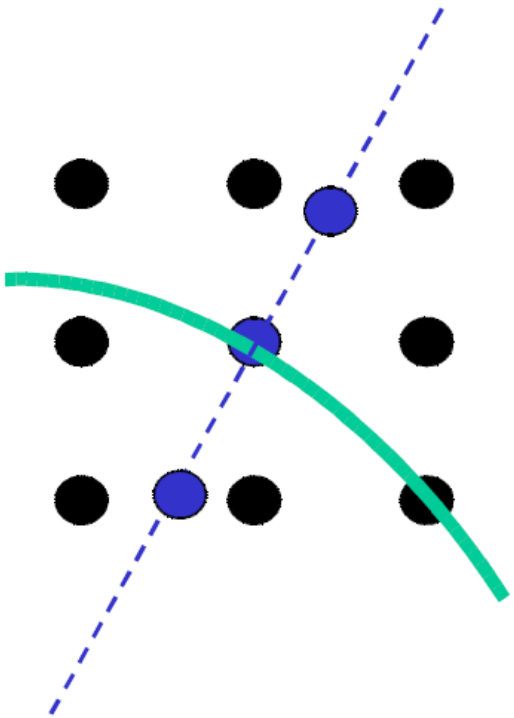
- Direction du gradient $\theta = \arctan(Gy / Gx)$
- Arrondi des directions par multiples de 45°



Filtre de Canny : étapes (2)

4 – Suppression des non-maxima

- Si la norme du gradient en un pixel (x,y) est inférieure à la norme du gradient d'un de ses 2 voisins le long de la direction du gradient, alors mettre la norme pour le pixel (x,y) à zéro

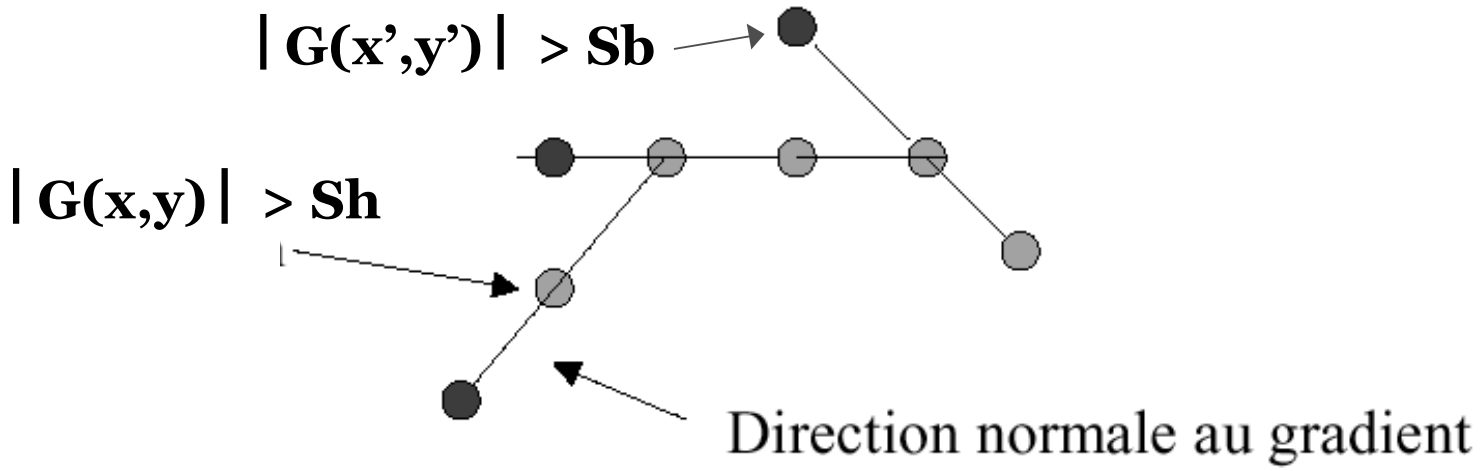


Source : Thomas Boudier, UPMC, www.snv.jussieu.fr/~wboudier/ens/cours_inb2/06_Segmentation.pdf

Filtre de Canny : étapes (3)

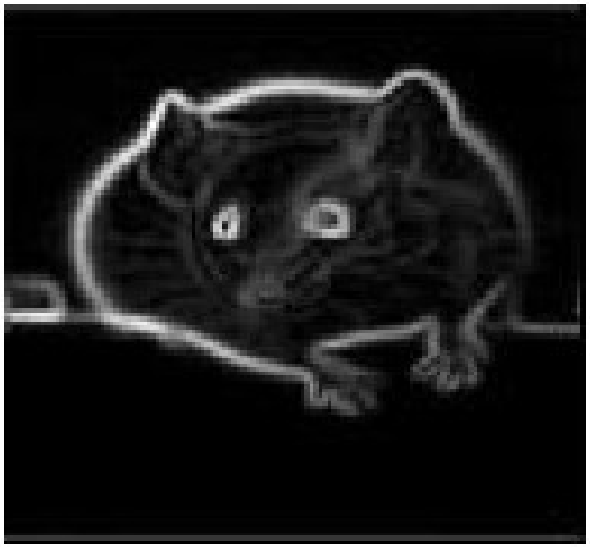
5 – Seuillage des contours (hystérésis)

- Utilise deux seuils : un seuil haut (Sh) et un seuil bas (Sb)
- *Pour chaque pixel de la norme du gradient :*
  - Si norme(x,y) < Sb, alors le pixel est mis à zéro (non-contour)
  - Si norme(x,y) > Sh, alors le pixel est contour
  - Si Sb ≤ norme(x,y) ≤ Sh, alors le pixel est contour s'il est connecté à un autre pixel déjà accepté comme contour



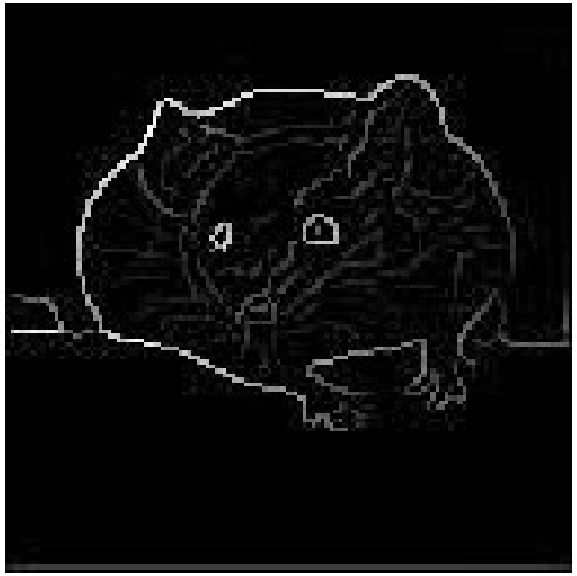
Filtre de Canny : exemple

Image originale



Sobel

Suppression des non-maxima



Seuillage

Source : Caroline Rougier. Traitement d'images (IFT2730). Univ. de Montréal.

Quel filtre choisir pour les contours ?

- Aucun opérateur n'est parfait pour détecter les contours
- En pratique, on obtient des contours incomplets
  - *il y a des pixels superflus*
  - *il y a des manques*
  - *il y a des erreurs de position et d'orientation des pixels contours*

- Chacun semble avoir sa préférence pour une méthode ou une autre
- Un opérateur de détection de contour n'est qu'une première étape dans la chaîne de segmentation

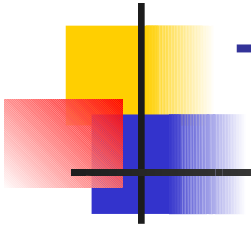
Approche globale pour les contours

Transformée de Hough (optionnel)

Approche globale pour les contours

- Il existe des **approches globales** pour les contours
  - On ne recherche pas seulement des pixels contours
  - On cherche le contour au complet
- On cherche plusieurs pixels correspondant à un contour
  - Comment définir le contour ? Problème ?
- Différentes techniques
  - Ici : Transformée de Hough

CVIPTools : Analysis>Edge/Line Detection>Hough



Transformée de Hough

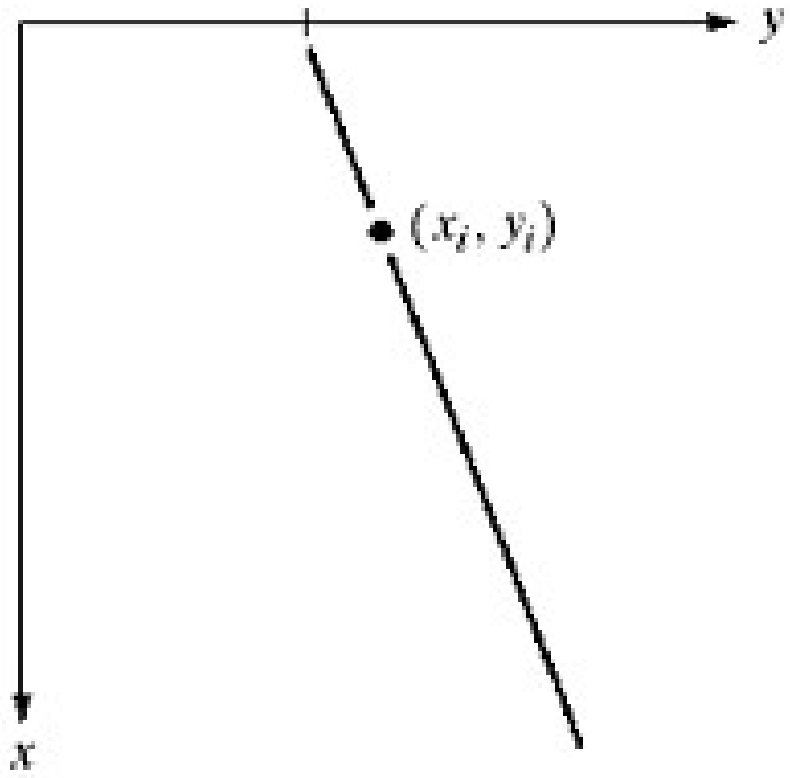
- Approche **globale** pour détecter des contours continus
  - *Du plan x-y au plan paramétrique a-b*

- *Plan x-y*
  - $y_i = a\, x_i + b$
  - Une infinité de lignes passent par (x_i, y_i)
  - Une seule ligne pour la paire (a,b)

- *Plan paramétrique a-b*
  - $b = - x_i a + y_i$
  - Une seule ligne pour la paire (x_i, y_i)
  - Une infinité de lignes passent par (a,b)

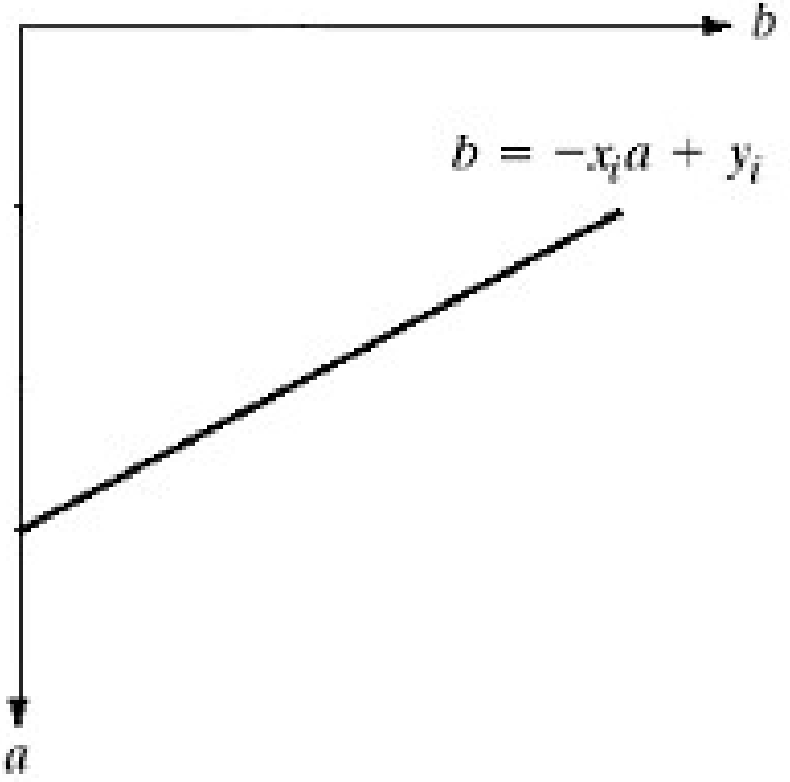
Plan x-y vs Plan a-b

Plan x-y



$y_i = a\ x_i + b$

Plan paramétrique a-b

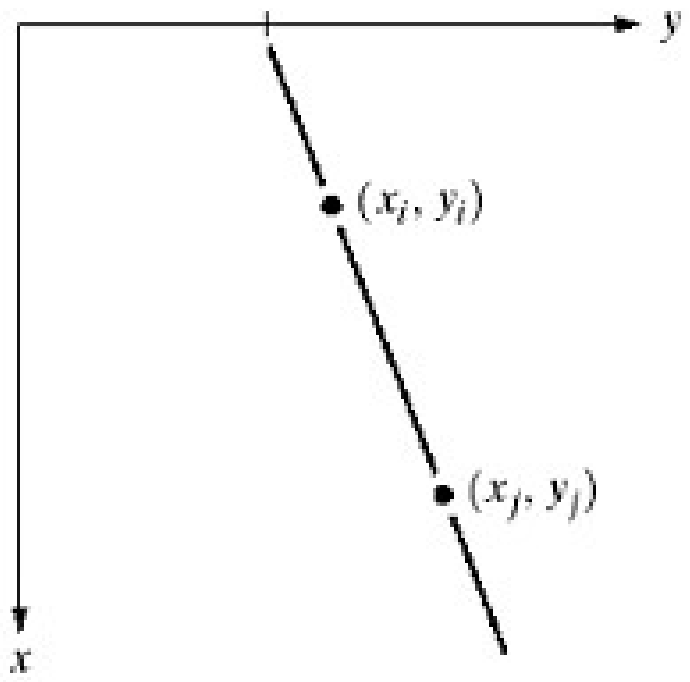


$b = -\ x_i\, a + y_i$

Droites vs Points

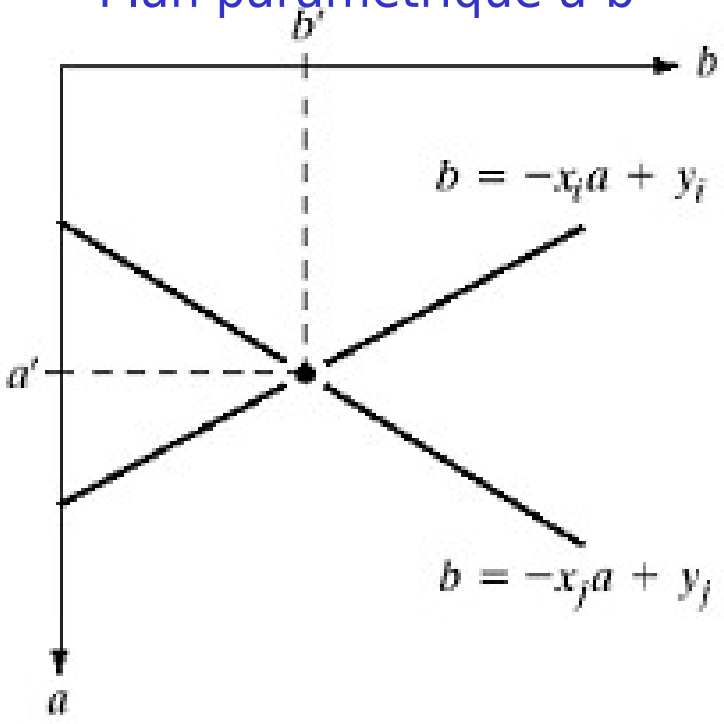
Tous les points (x,y) sur une ligne du plan x-y passent par un seul point (a', b') dans le plan paramétrique a-b

Plan x-y



$y_i = a\ x_i + b$

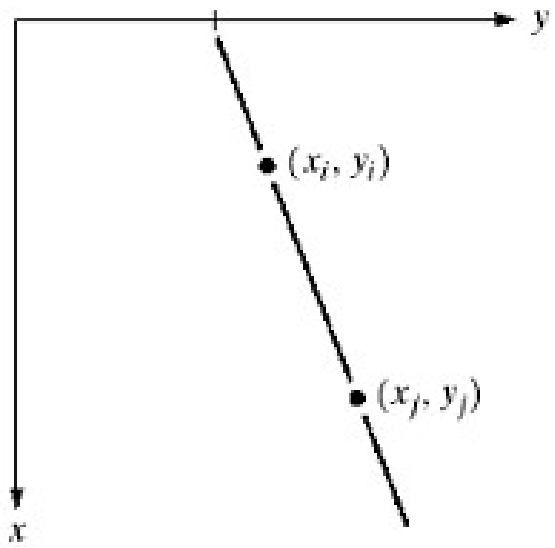
Plan paramétrique a-b

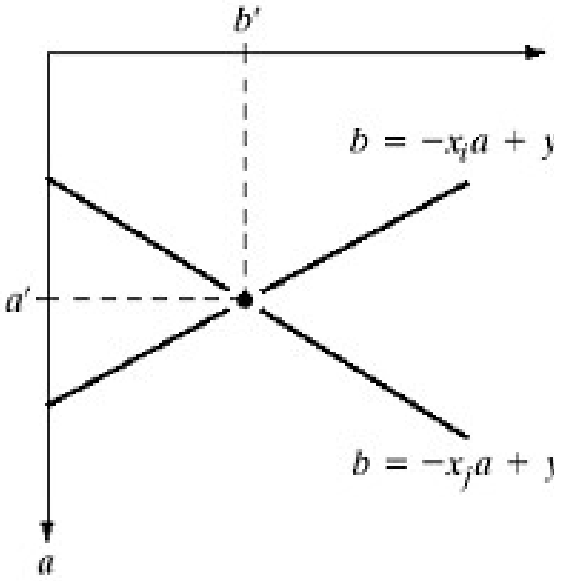


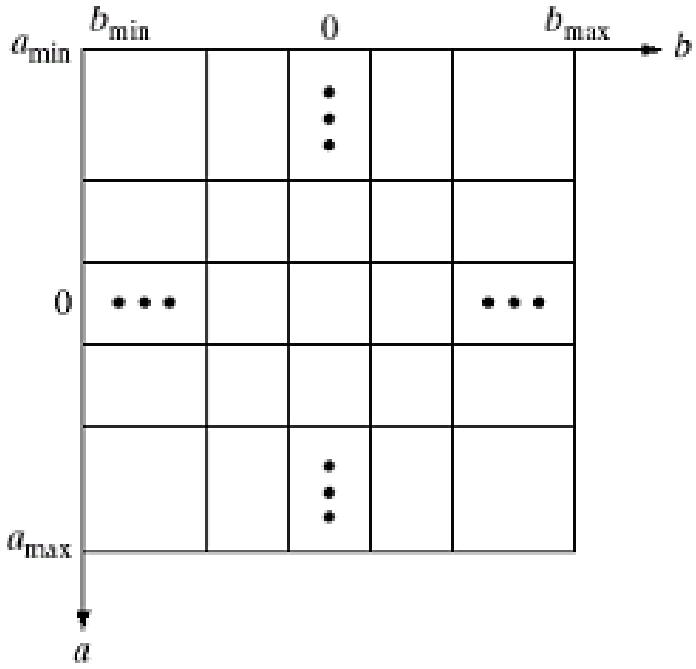
$b = - x_i\, a + y_i$

Principe de la transformée de Hough

- Cellules d'accumulation - Matrice (a,b)
- On construit une **image des votes**
  - *chaque point permet de voter pour une droite particulière*
- Les droites recevant le plus de votes sont conservées







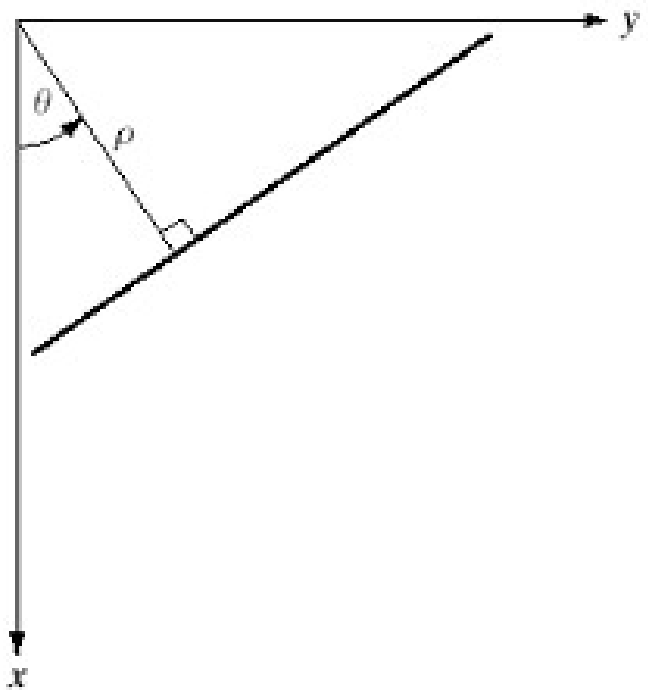
Calcul de la transformée de Hough

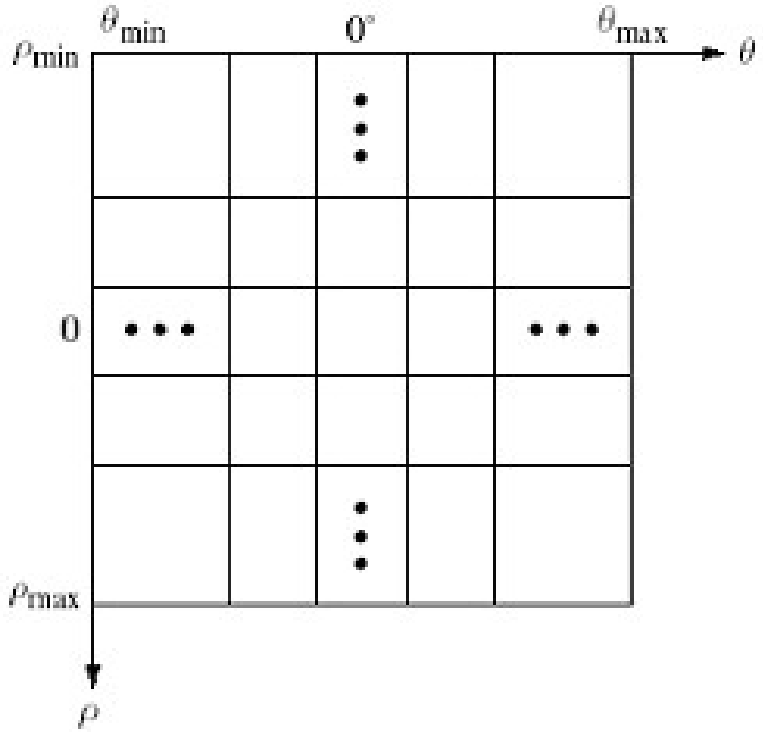
- On calcule le gradient de l'image originale
  - *Sobel, Prewitt, Canny, …*
- Pour chaque point du gradient, on calcule une droite (a,b)
  - *On obtient une droite dans le plan a-b pour chaque pixel (x,y)*
- Les pics maximum dans le plan paramétrique a-b indiquent les droites avec le maximum de points du plan x-y
  - *Les points de croisement des droites dans le plan a-b indiquent les vraies droites existantes dans le plan x-y*

Problème avec un espace (a,b)

- Problème : pour une droite verticale, b=∞ !
- Solution : représentation sous forme polaire (ρ,θ)

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





θ = ±90

Exemple avec 5 points

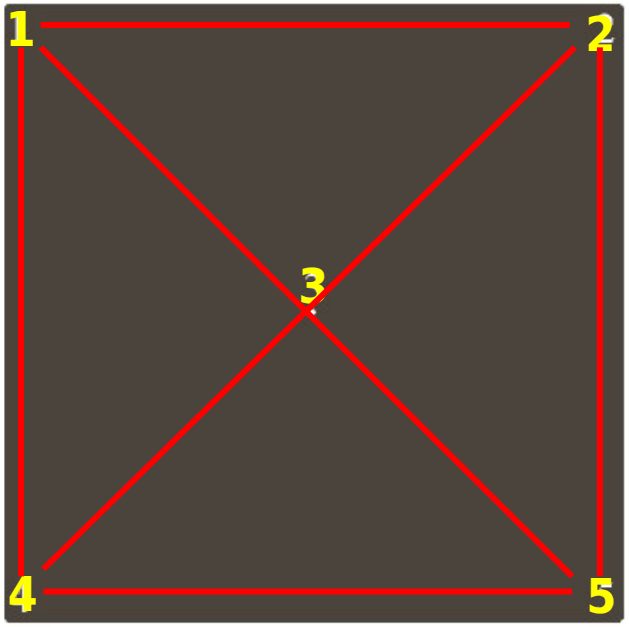
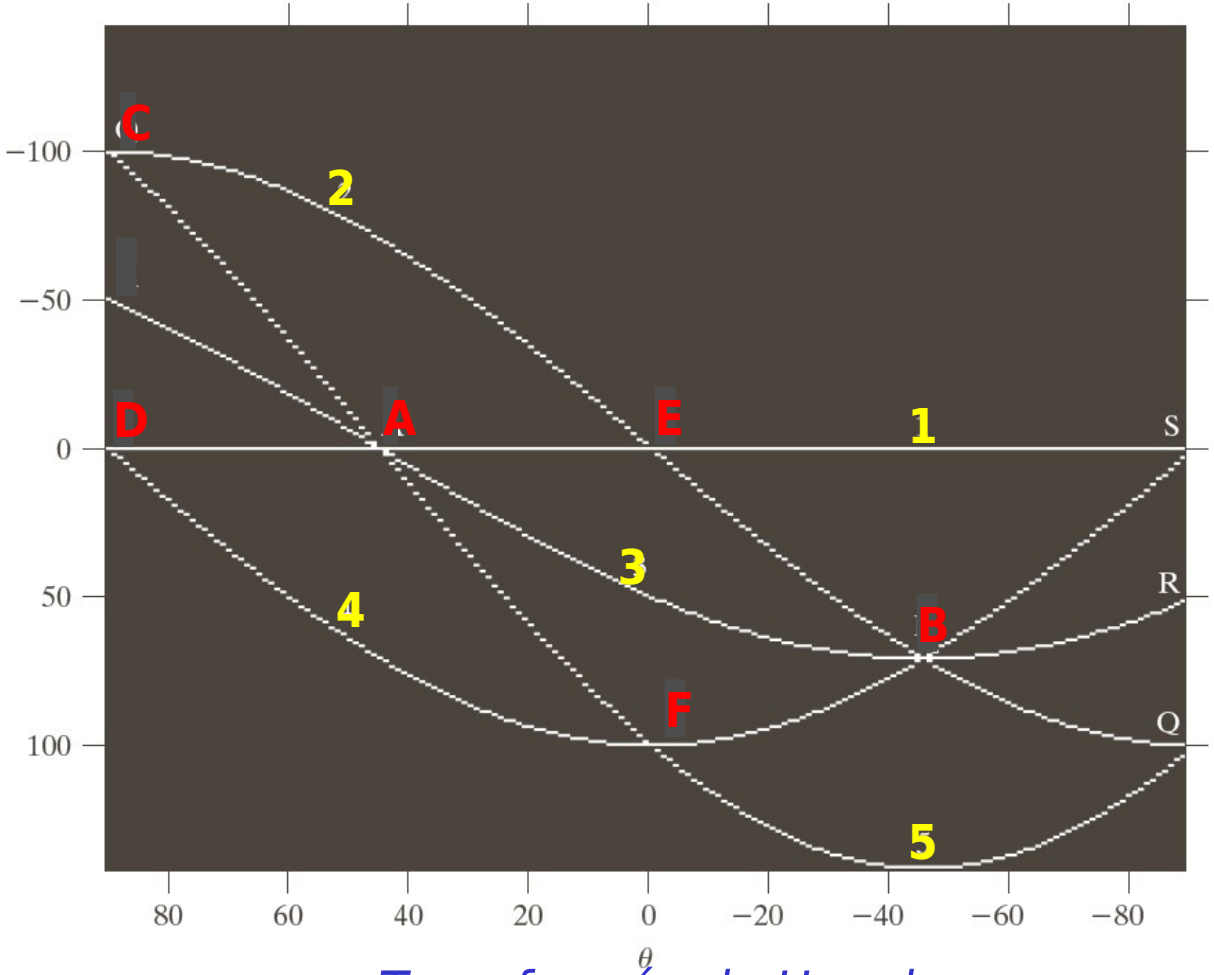


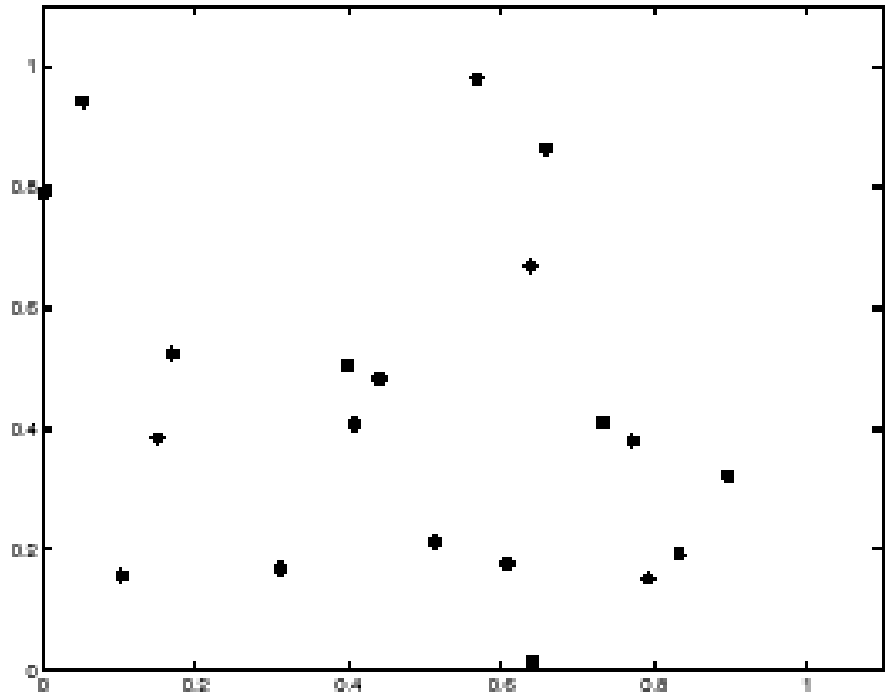
Image contenant 5 points

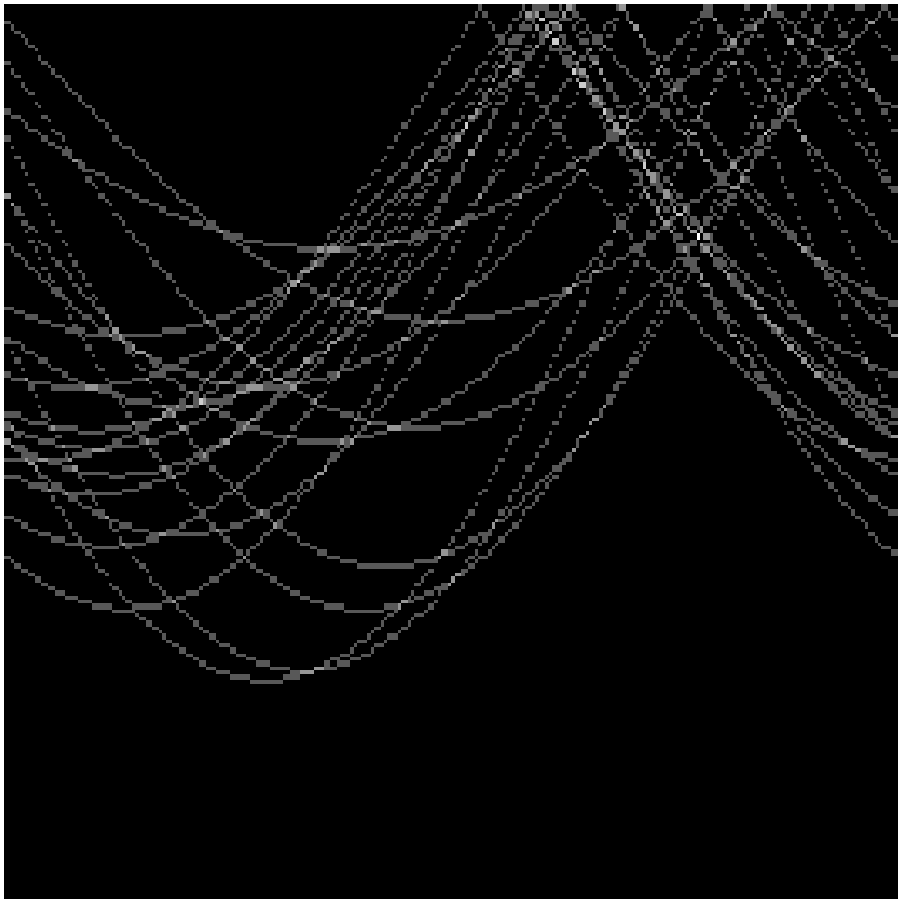


Transformée de Hough

Source : Gonzalez and Woods. Digital Image Processing 3ed. Prentice-Hall, 2008.

Transformée de Hough (points)

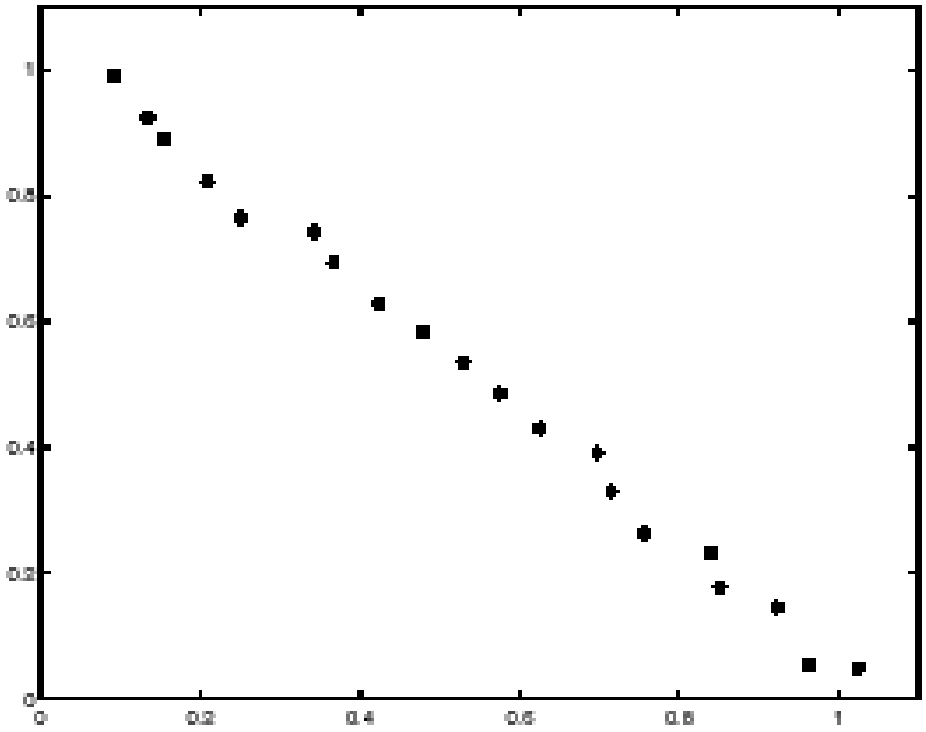


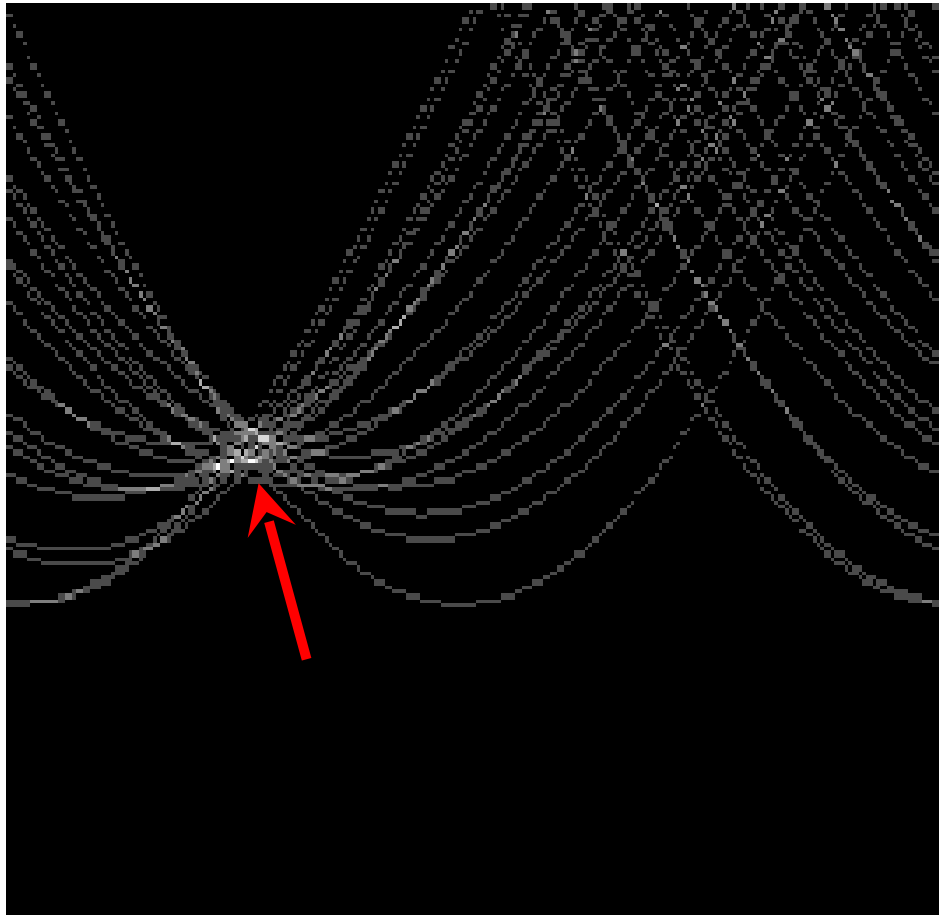


La transformée de points aléatoires ne donne rien de précis

Transformée de Hough (droite)

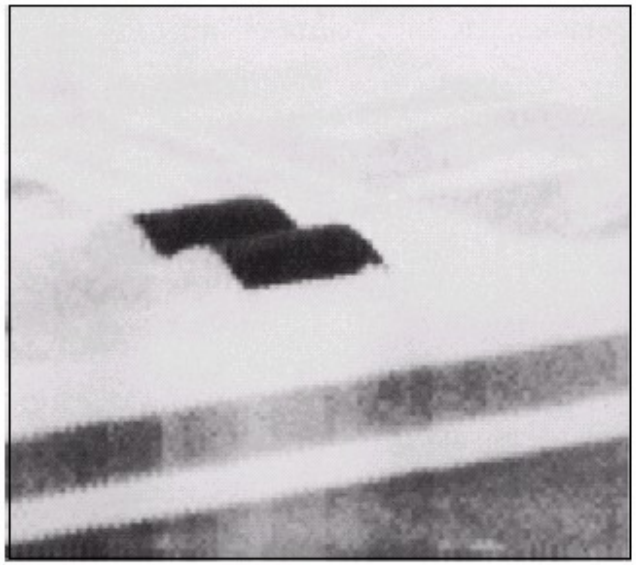
La transformée de points alignés permet de retrouver la droite

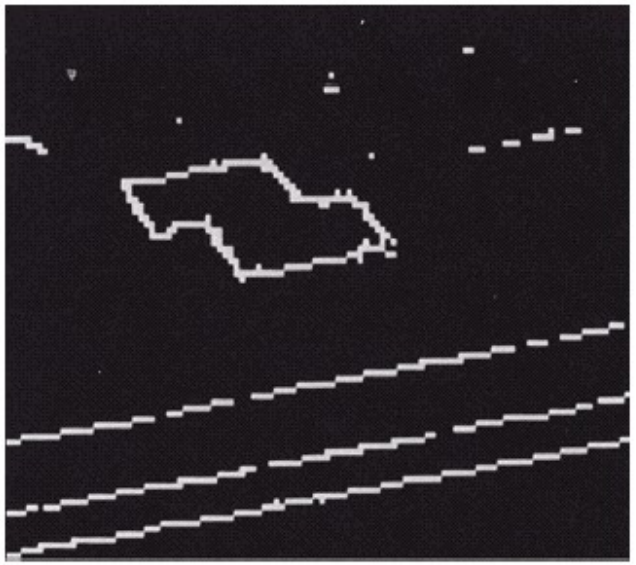




Exemple

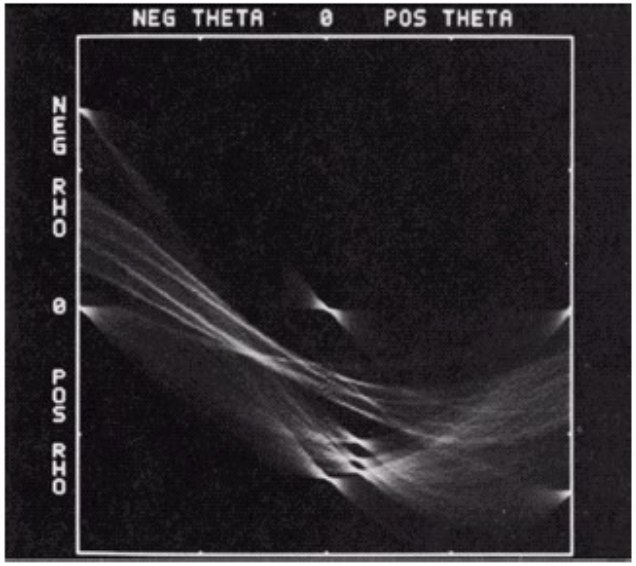
Image

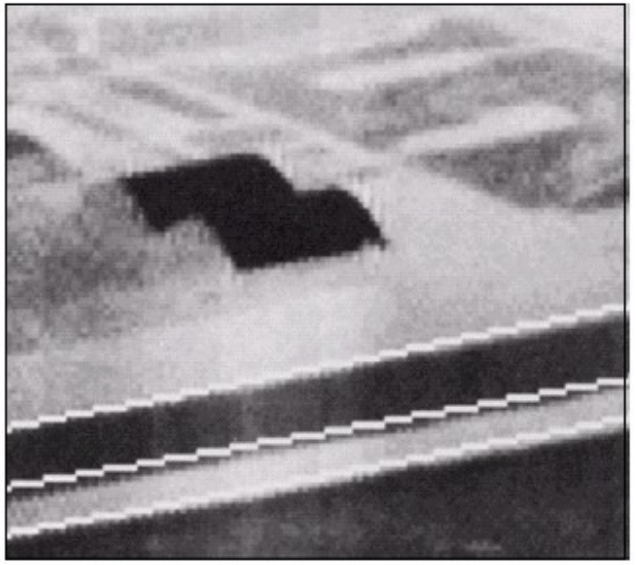




Gradient

Hough





Final

Source : Gonzalez and Woods. Digital Image Processing. Prentice-Hall, 2002.

Autre exemple

Image

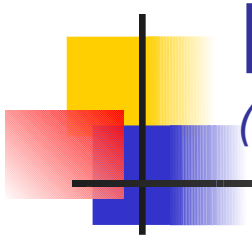
Canny

Hough

Droites

Final

Source : Gonzalez and Woods. Digital Image Processing. Prentice-Hall 3ed, 2008.



Références

(voir aussi la page web du cours)

- Gradient. Wikipédia (fr). (page consultée le 2 avril 2009).
  - http://fr.wikipedia.org/wiki/Gradient

- Caroline Rougier. Cours de Traitement d'images (IFT2730). Université de Montréal (Canada)
  - http://www-etud.iro.umontreal.ca/~rougierc/ift2730/
  - Chap10 : Filtrage : lissage, réhaussement d'images, détection de contours : http://www-etud.iro.umontreal.ca/~rougierc/ift2730/cours/Cours10_IFT2730_2008_2.pdf