Détection de Contours
Master Système Embarqué

Aissa. Belmeguenai

Laboratoire de Recherche en Electronique de Skikda
Université 20 Août 1955- Skikda
BP 26 Route d'El-hadaeik Skikda, Algeria

2020

Plan du cours

1 Introduction

La détection de contours est une technique de réduction d'information dans les images, qui consiste à transformer l'image en un ensemble de courbes, pas forcément fermées, formant les frontières significatives de l'image.

Définitions

- Les contours de l'image représentent:
- Discontinuités de l'image
- Frontières entre deux régions homogènes adjacentes ayant des intensités lumineuses différentes.
- Frontière qui sépare deux objets (ou 1 objet du fond) dans une image.
- Les contours forment des segments, courbes, motifs
- Le contour permet la réduction de l'information:
  - Information de toute l'image résume dans les contours des différent objets,
  - Contour : partie la plus informative d'une image,

Applications:

- Reconnaissance d'objets, de formes, classifications de scènes;
- Mise en correspondance: calibration, reconstruction 3D;
- Compression;
- Trés utile pour toute application de traitement d'image.

2 Modèle de contour

- Il existe plusieurs modèles de contours, le modèle le plus courant: Marche d'escalier;
- Exemple de différents modèles de contours: marche d'escalier, rampe et toit:

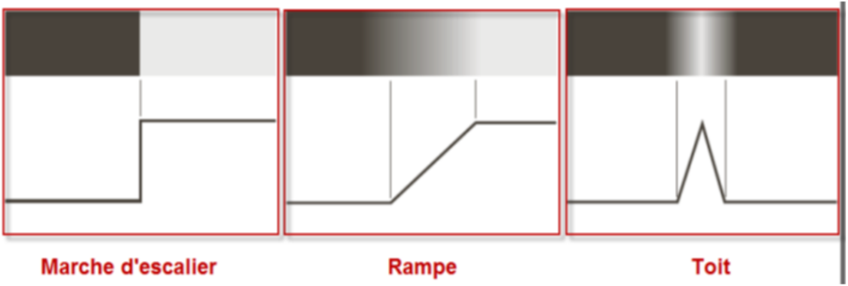


Figure : *Modèles des contours*

2 Modèle de contour

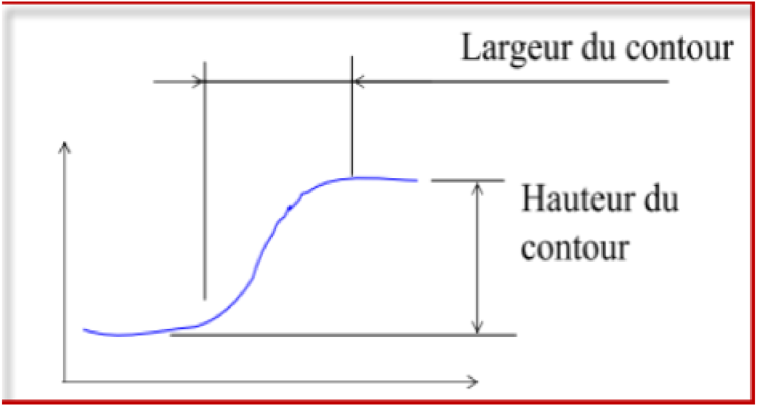


Figure : *Largeur et Hauteur du contour*

3. Des dérivées aux points contours

Dérivées première et seconde:

- Pour détecter les contour de l'image on utilise:
  - Dérivée première : **maxima locaux**
  - Dérivée seconde : passages par**zéro**

Cas 1-D

- Dérivée première : Le contour corresponde donc à **un maximum** de la dérivée première, $\frac{\partial f}{\partial x}$ est maximal.
- Dérivée seconde : Le contour corresponde donc à un passages par**zéro** de la dérivée seconde, $\frac{\partial^2 f}{\partial x^2} = 0$.

Cas 1-D

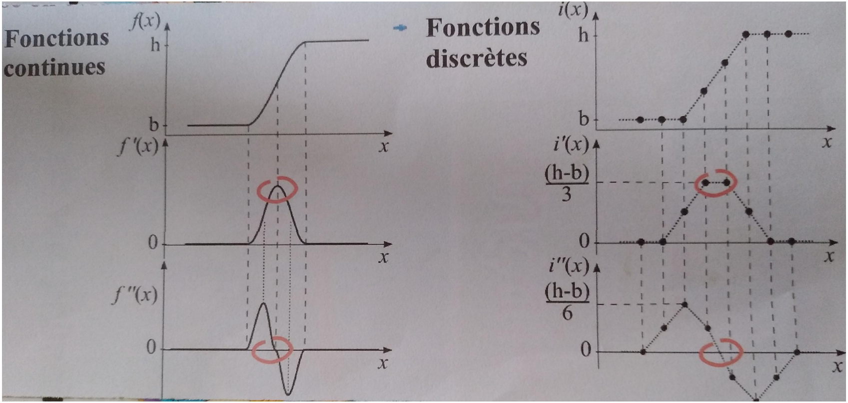


Figure : *Dérivée première et dérivée seconde 1-D*

Cas 2-D

- Dérivée première : Les contours correspondent donc aux maximas du gradient dans la direction du gradient, $\overrightarrow{\nabla} f(x, y) = \left(\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \right)$
- Dérivée seconde : Les contours correspondent donc aux ou les passages par zros du Laplacien,$\triangle f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$.

Cas 2-D

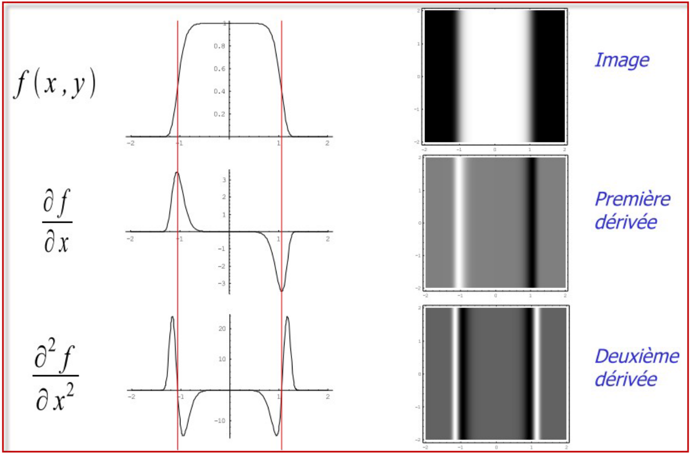


Figure : *Dérivée première et dérivée seconde 2-D*

4. Approches de détection de contour

Deux approches sont les plus utilisées:

- Approches par dérivée premières: Approximations du Gradient;
- Approches par dérivée secondes: Approximations du Laplacien.

Plan du cours

4.1 Approches par dérivées premières: Approximations du gradient

Principe de détection de contour par le gradient:

Soit $f(x, y)$ une image.

1. Calcule les dérivées partial:
   $\frac{\partial f(x,y)}{\partial x}$ et $\frac{\partial f(x,y)}{\partial y}$
2. Calcul de la norme du gradient:
   $$\|\overrightarrow{\nabla} f(x, y)\| = \sqrt{\left(\frac{\partial f(x,y)}{\partial x}\right)^2 + \left(\frac{\partial f(x,y)}{\partial y}\right)^2}$$
3. Seuillage: extrait les points contours
   - Les points contours : $f_{contour} = \begin{cases} 1 & si \|\overrightarrow{\nabla} f\| \geqslant seuil \\ 0 & sinon \end{cases}$

Principe de gradient

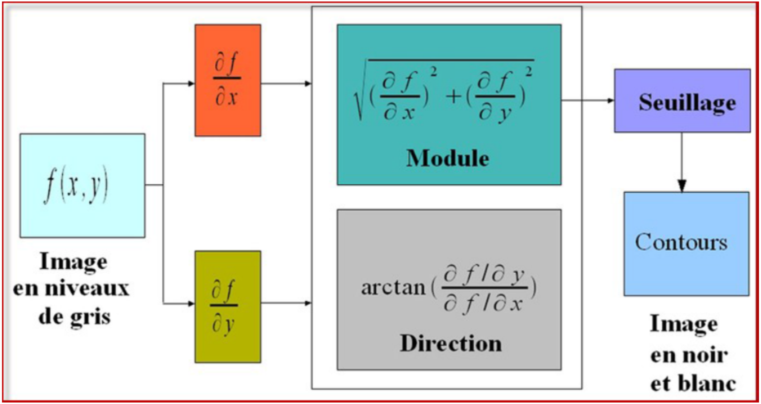


Figure : *Principe du gradient*

Gradient

Dans le cas 2D des images, le vecteur gradient est défini au point de coordonnes (x, y) par:

$$\vec{\nabla} f(x,y) = \left(\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \right) \tag{1}$$

Où $f(x, y)$ reprsente l'image.

- Le gradient peut être représente en coordonnées polaires par un module et une direction dans l'image.
- Le module du gradient mesure la force du contour est donné par:

$$\|\vec{\nabla} f(x,y)\| = \sqrt{\left(\frac{\partial f(x,y)}{\partial x}\right)^2 + \left(\frac{\partial f(x,y)}{\partial y}\right)^2} \tag{2}$$

- La direction du gradient est:

$$\theta(x,y) = \arctan\left(\frac{\partial f(x,y)/\partial y}{\partial f(x,y)/\partial x}\right) \tag{3}$$

Gradient

- Le gradient est un vecteur perpendiculaire au Contour
- Un contour apparait comme une ligne où sont localises les trés fortes variations de $f(x, y)$.
- Un contour est alors défini comme le lieu des maximas de la dérivée premiére dans la direction du gradient.

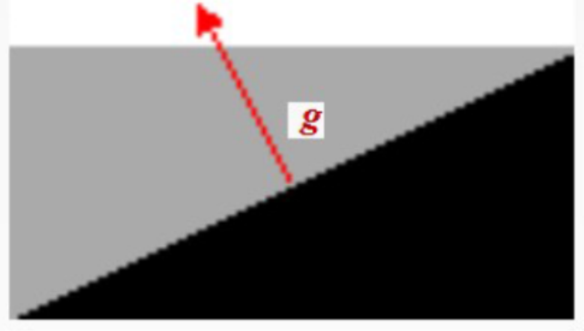


Figure : *Gradient*

Approximation discrète du Gradient

- Pour estimer le gradient, on choisit deux directions privilégiées orthogonales (les lignes et les colonnes) sur lesquelles on le projette
- Le module du gradient en un point (x, y) est donné par:

$$G(x,y) = \|\overrightarrow{\nabla} f(x,y)\| = \sqrt{(G_x(x,y))^2 + (G_y(x,y))^2} \tag{4}$$

- La direction du gradient en un point (x, y) est donne par :

$$\theta(x,y) = \arctan\left(\frac{G_y(x,y)}{G_x(x,y}\right) \tag{5}$$

Approximation discrète du Gradient

Cas 1:

- $\frac{\partial f(x,y)}{\partial x} = G_x(x,y) \approx f(x,y) - f(x-1,y) = f(x,y) * h_x$
  Ce qui correspond à une convolution avec le masque: $h_x = \begin{bmatrix} 1 & -1 \end{bmatrix}$
- $\frac{\partial f(x,y)}{\partial y} = G_y(x,y) \approx f(x,y) - f(x,y-1) = f(x,y) * h_y$
  Ce qui correspond à une convolution avec le masque: $h_y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$
- La norme du gradient en un point (x,y) est donne par:
  $G(x,y) = \sqrt{(f(x,y) - f(x,y-1))^2 + (f(x,y) - f(x-1,y))^2}$

Approximation discrète du Gradient

Cas 2:

- $\frac{\partial f(x,y)}{\partial x} = G_x(x,y) \approx f(x+1,y) - f(x-1,y) = f(x,y) * h_x$
  Ce qui correspond à une convolution avec le masque:
  $h_x = \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$
- $\frac{\partial f(x,y)}{\partial y} = G_y(x,y) \approx f(x,y+1) - f(x,y-1) = f(x,y) * h_y$
  Ce qui correspond à une convolution avec le masque: $h_y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix}$
- La norme du gradient en un point (x,y) est donnée par:
  $G(x,y) =$
  $\sqrt{(f(x,y+1) - f(x,y-1))^2 + (f(x+1,y) - f(x-1,y))^2}$

Masque gradient

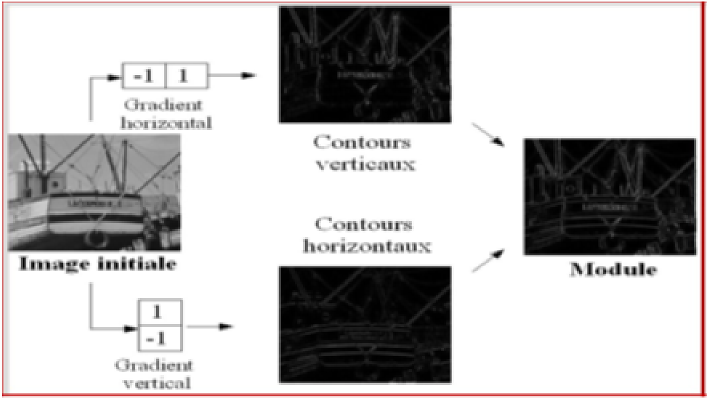


Figure : *Exemple d'application des masques gradient*

Filtres dérivatifs du premier ordre

- Un filtre est composé de deux masques:
  - Le masque h_x Détecte les contours verticaux
  - Le masque h_y Détecte les contours horizontaux
- Dans le domaine spatial: convolution linéaire de l'image avec les filtres
  - $G_x = f * h_x$: Image des contours verticaux, ou gradient vertical
  - $G_y = f * h_y$: Image des contours horizontaux, ou gradient horizontal
- Module du gradient:
  $G(x, y) = \sqrt{(G_y)^2 + (G_x)^2}$
- Filtres courants : Roberts, Prewitt, Sobel.

Opérateurs du premier ordre

- Opérateur de Roberts : $h_x = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ et $h_y = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$
- Opérateurs de Prewitt:
  $h_x = \begin{pmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{pmatrix}$ et $h_y = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$
- Opérateurs de Sobel:
  $h_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$ et $h_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix}$

Opérateurs du premier ordre

- Ces masques opèrent tout d'abord un lissage de l'image, suivi d'une opération de dérivation,
- Ils sont dits séparables:

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

et $h_y = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}.$

- On combine à la fois un filtrage et une dérivée: filtrage passe bas (intégrateur) + filtrage passe haut (différentiateur).
- Moins sensible au bruit que le calcul direct des dérivées.
- Prewitt $\Leftrightarrow$ filtre moyenneur +Dérivées.
- Sobel $\Leftrightarrow$ filtre gaussien + Dérivées.

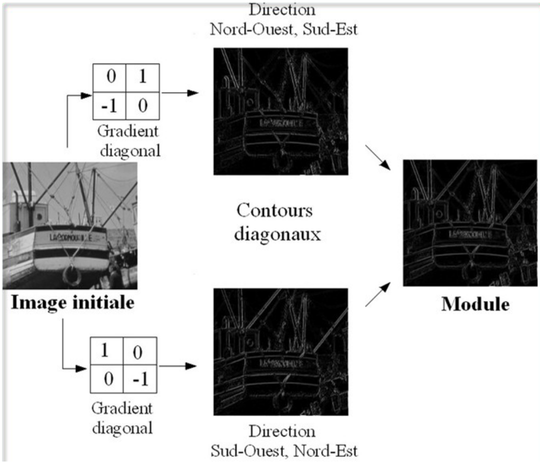


Figure : *Détection de contour par le masque de Roberts*

Exemple d'application des opérateurs du premier ordre

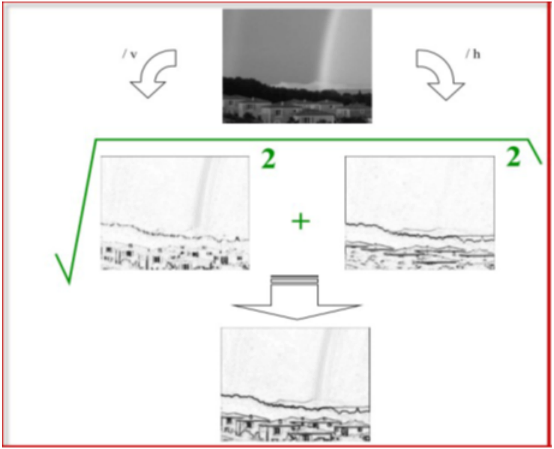


Figure : *Détection de contour par le masque de Sobel*

Exemple d'application des opérateurs du premier ordre

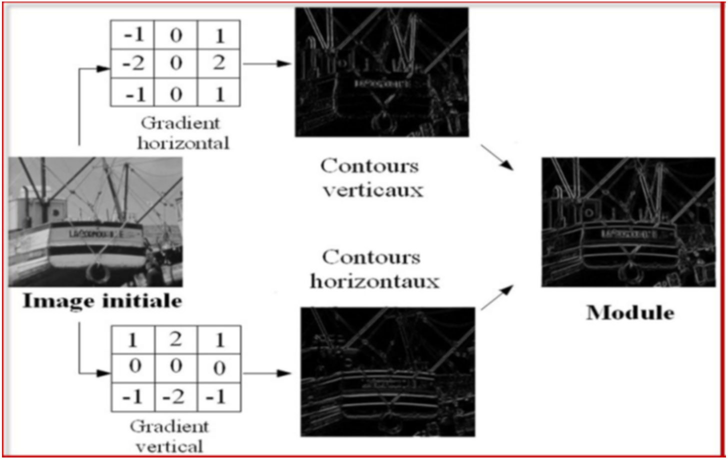


Figure : *Détection de contour par le masque de Sobel*

Exemple d'application des opérateurs du premier ordre

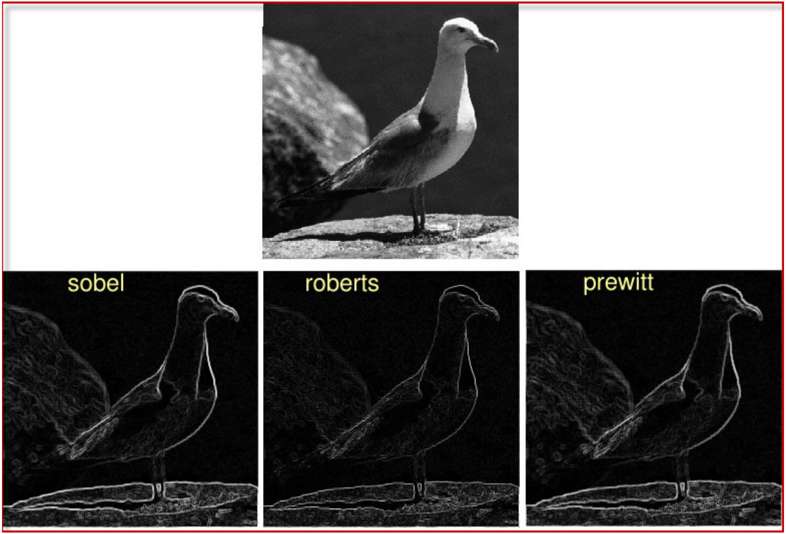


Figure : *Détection de contour par Sobel, Roberts Perwitt*

Seuillage

- Le seuillage est le traitement permettant de sélectionner les contours les plus significatifs dans l'image représentant le module du gradient.
- Une fois la norme du gradient $G(i,j)$ calculée en chaque point de l'image, il faut seuiller cette norme pour décider si un pixel fait partie ou non d'un contour.
- Cette opération nécessite le réglage d'un paramètre: le **seuil S**.

$$G(i,j)_{contour} = \begin{cases} 1 & si G(i,j) \geqslant S \\ 0 & sinon \end{cases}$$

(6)

Seuillage

Tout le problème réside alors dans le choix du seuil :

- Seuil bas (faible lissage): contours pertinents détectés mais beaucoup de bruit.
    - Sensibilité au bruit
    - Bonne détection
- seuil haut (fort lissage) : nombreux contours manqués mais absence de bruit
    - Robustesse au bruit
    - Mauvaise détection
- Il est difficile de choisir un seuil optimal pour toute l'image, à cause du bruit, variation de luminance, du contraste.

Seuillage par Hystérésis :

L'image contour est une image binaire des pixels contours (0=non contour, 1=contour).

- On introduit maintenant 2 seuils : un seuil haut S_h et un seuil bas S_b.
- Soit C l'image contour de même taille que G l'image de la norme du gradient, pour chaque pixel (i,j),alors:
- Si la norme du gradient $G(i,j) > S_h$, $C(i,j) = 1$ le pixel (i,j) est contour.
- Si la norme $G(i,j) < S_b$, $C(i,j) = 0$ le pixel (i,j) n'est pas contour.
- Si $S_b < G(i,j) < S_h$, $C(i,j) = 1$ s'il est connecté à un autre pixel déjà contour.

Exemple de seuillage

Image du gradient | Seuillage fixe à 50 | Seuillage fixe à 80

Figure : *Détection de contour par le seuillage de l'image gradient.*

Exemple de seuillage

Figure : *Détection de contour par le masque de Sobel avec un seuillage.*

Plan du cours

4.2 Approximations du Laplacien

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.

- Aux points contours, la dérivée seconde est nulle.
- Plus précisément, les points contours sont caractérisés par un **passage par zéro** zero crossing de la dérivée seconde.

L'opérateur de Laplacien

L'opérateur de Laplacien, défini par :
$\triangle f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$
Remarque :le Laplacien est un opérateur scalaire et isotrope (invariant par rotation).

Utilisation du Laplacien pour la détection des points contours

- Les passages par zéro de la dérivée seconde directionnelle f'' coïncident généralement avec ceux du Laplacien $\triangle f = 0$
- En particulier: (x_0, y_0) est point contour si

$$\triangle f(x_0, y_0) = \frac{\partial^2 f(x_0, y_0)}{\partial x^2} + \frac{\partial^2 f(x_0, y_0)}{\partial y^2} = 0 \tag{7}$$

- Un passage par zéro du Laplacien correspond donc à une dérivée seconde directionnelle nulle dans la direction du gradient.

L'opérateur de Laplacien

Dérivées secondes discrètes

- $\triangle f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$
- $= \frac{\partial}{\partial x}[f(x+1,y) - f(x,y)] + \frac{\partial}{\partial y}[f(x,y+1) - f(x,y)]$
- $\frac{\partial}{\partial x}[f(x+1,y) - f(x,y)] =$ $[f(x+1,y) - f(x,y)] - [f(x,y) - f(x-1,y)]$
- $= f(x+1,y) - 2f(x,y) + f(x-1,y)$
- Masque associé à la dérivée seconde selon x : $\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$.

Lopérateur de Laplacien

- De même
- $\frac{\partial}{\partial y}[f(x,y+1)-f(x,y)] =$
  $[f(x,y+1)-f(x,y)]-[f(x,y)-f(x,y-1)]$
- $= f(x,y+1)-2f(x,y)+f(x,y-1)$
- Masque associé à la dérivée seconde selon y : $\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$.

- D'où l'approximation discrète du Laplacien
- $\triangle f(x,y) = f(x+1,y)+f(x-1,y)-4f(x,y)+f(x,y+1)+f(x,y-1)$
- Qui se calcule donc grâce au masque de convolution : $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$.

L'opérateur de Laplacien

- Autres approximations possibles du Laplacien :
- $\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$.
- $\begin{bmatrix} 1 & 2 & 1 \\ 1 & -12 & 1 \\ 1 & 2 & 1 \end{bmatrix}$.
- $\begin{bmatrix} 1 & 4 & 1 \\ 1 & -20 & 1 \\ 1 & 4 & 1 \end{bmatrix}$.

L'opérateur de Laplacien

Détection des passages par zéro du laplacien

En Pratique:

- Prendre une fenètre centrée 3×3 sur le pixel (n, m), et calculer $max(\triangle f)$ et $min(\triangle f)$ dans ce voisinage.
- Le passage par 0 sera détecté si $max(\triangle f) > 0$, $min(\triangle f) < 0$ et $max(\triangle f) - min(\triangle f) > S$. Où $\triangle f$ est Laplacien de l'image f.

Passages par zéro

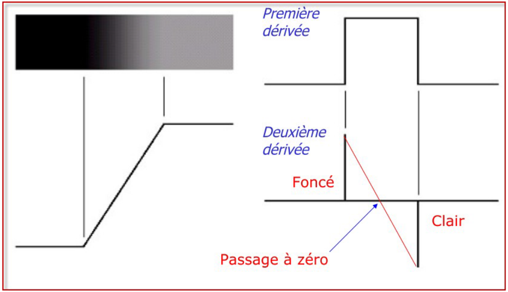


Figure : *Passages par zéro*

Exemples dimages

Figure : *Exemple dapplication de le Laplacien*

Exemples de détection de contour dimages par le Laplacien avec seuillage

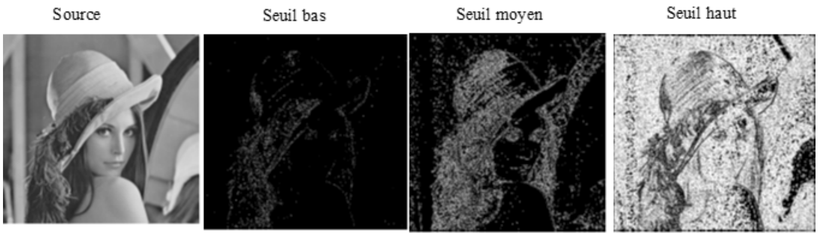


Figure : *Exemple d'application de le Laplacien avec seuillage*

Réduction de la sensibilité au bruit

- Du fait du double dérivation, le Laplacien est trés sensible au bruit.
- On préfère utiliser le Laplacien à partir de dérivées de gaussiennes
- Avant d'utiliser le Laplacien pour détecter les points contours.
- Possibilité de réaliser ces deux opérations en une seule :
- Gaussien définie par:
  $g(x, y) = \frac{1}{2\pi\sigma^2} \exp(-\frac{x^2+y^2}{2\sigma^2})$
- $\Delta\left[(g * f)\right] = (\Delta g) * f = LoG * f$

Réduction de la sensibilité au bruit

Expression du l'opérateur LoG

- $\frac{\partial g(x,y)}{\partial x} = \frac{x}{2\pi\sigma^4} \exp(-\frac{x^2+y^2}{2\sigma^2})$
- $\frac{\partial^2 g(x,y)}{\partial x^2} = \frac{1}{2\pi\sigma^4} \left(1 - \frac{x^2}{\sigma^2}\right) \exp(-\frac{x^2+y^2}{2\sigma^2})$
- $\frac{\partial g(x,y)}{\partial y} = \frac{y}{2\pi\sigma^4} \exp(-\frac{x^2+y^2}{2\sigma^2})$
- $\frac{\partial^2 g(x,y)}{2} = \frac{1}{2\pi\sigma^4} \left(1 - \frac{y^2}{\sigma^4}\right) \exp(-\frac{x^2+y^2}{2\sigma^2})$
- D'où:
  $LoG(x,y) = \triangle g(x,y) = \frac{1}{2\pi\sigma^4} \left(1 - \frac{x^2+y^2}{\sigma^4}\right) \exp(-\frac{x^2+y^2}{2\sigma^2})$

Approximation de LoG par différence de gaussiennes (DoG)

Le filtre LoG peut être approché par la diffrence de deux gaussiennes d'écarts-types proches :

- $$LoG(x, y) = DoG(x, y, \sigma_1, \sigma_2) = g(x, y, \sigma_1) - g(x, y, \sigma_2) \quad (8)$$

  Avec $\sigma_1 = \sigma_2 + \epsilon$

Localiser les contours aux passages par 0 de LoG

Algorithme

- Convolution de l'image avec LoG (approché par un masque de taille $n \times m$);
- Autre possibilité: utiliser $DoG(x, y, \sigma_1, \sigma_2)$ (avantage : filtres gaussiens séparables);
- Détection des passages par 0 de l'image résultante (éventuellement :) seuillage des passages par 0.

Détection de contours par l'opérateur LoG

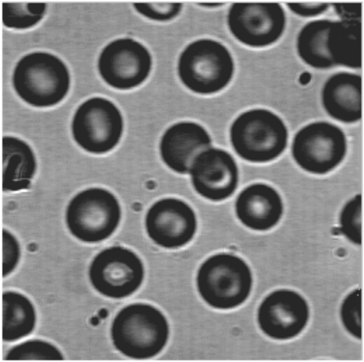


Image Originale

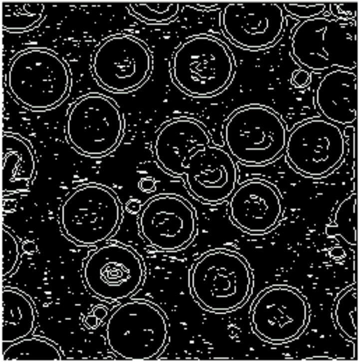
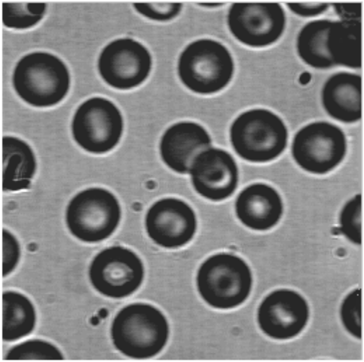


Image des contours
(opérateur LoG avec $\sigma=1$)

Figure : *Détection de contours par l'opérateur LoG*

Détection de contours par l'opérateur LoG



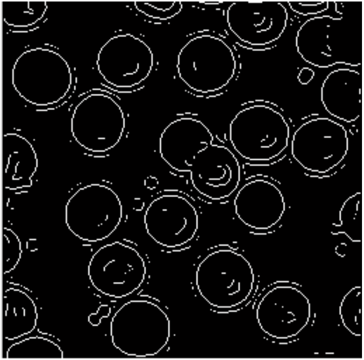


Image originale

Image des contours
(opérateur LoG avec σ=2)

Figure : *Détection de contours par l'opérateur LoG*

Avantages et Inconvénients

Avantages:

- Proche du système visuel humain
- La détection des passages par 0 fournit des réseaux de lignes fermées
- Prise en compte des seuls passages par 0 significatifs.

Inconvénients:

- Grande sensibilité au bruit (nécessite un lissage fort, affecte la localisation)
- Pas d'info sur l'orientation du contour
- Le seuillage des passages par 0 crée des lacunes (ouvertures) dans les contours.

Limites: Pour σ élevé, localisation médiocre et points d'intérêt saillants perdus .

5. Conclusion

- Pas d'opérateur parfait pour détecter les contours
- On obtient en pratique des contours incomplets (ouverts)
  - Détection incorrecte : pixels superflus, pixels manquants
  - Localisation incorrecte : erreurs dans la position des points contours, l'orientation

  - La détection des points contours n'est que la première étape dans la chaîne de segmentation.

References

[Khalid HOUSNI]
Cours de traitement dimage
https://docplayer.fr/48500117Coursdetraitementd'imageprkhalidhousnidepartementinformatiquefacultedesscienceskenitra.html

[N. Thome, D. Brziat, S. Dubuisson], (2016)
Bases du traitement des images: Détection de contours
http://webia.lip6.fr/~thomen/Teaching/BIMA/cours/contours.pdf

[Olivier Losson]
Cours:TI Traitement d'Images Semaine 9 : Détection de contours (1)
http://master-ivi.univ-lille1.fr/fichiers/Cours/ti-semaine-09-contours1.pdf

[Olivier Losson]
Cours:TI Traitement d'Images Semaine 10 : Détection de contours (2)
http://master-ivi.univ-lille1.fr/fichiers/Cours/ti-semaine-10-contours2.pdf