

Formation et Analyse d'Images

James L. Crowley

ENSIMAG 3

Premier Sémestre 2009/2010

Séance 6

16 novembre 2009

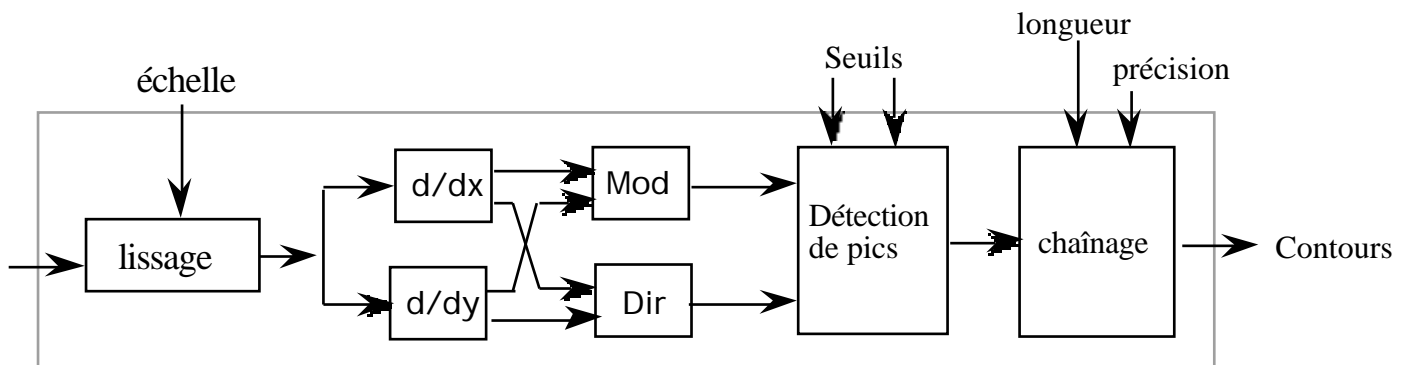
Détection et Description de Contraste

Plan de la Séance :

Détection de contrastes par dérivées.....	2
Détection des Pics dans le module du Gradient.....	3
Transformée de Hough.....	4
Généralisation de Transformée de HOUGH.....	5
Les operateurs Differentielles.....	6
Les Deuxièmes dérivées.....	7
Passages à zéro dans la dérivée seconde.....	9
Les operateurs Differentielles.....	10
La fonction Gaussien.....	11
Les dérivées de la Gaussien.....	11
Les Filtres Numérique Gaussien.....	12
Les Dérivées de l'Image.....	14

Détection de contrastes par dérivées

Schéma du processus de description de contraste



$$P(i, j) = \frac{\frac{p(i,j)}{i}}{\frac{p(i,j)}{j}} = \frac{m_i * p(i,j)}{m_j * p(i,j)} = \frac{E_i(i,j)}{E_j(i,j)}$$

$$\text{Module} = \| P(i, j) \| = E(i, j) = \sqrt{E_i(i, j)^2 + E_j(i, j)^2}$$

$$\text{Orientation} = (i, j) = \text{Tan}^{-1}\left(\frac{E_j(i, j)}{E_i(i, j)}\right)$$

Étapes :

- 1) Lissage : suppression du bruit, conditionnement du signal
- 2) Calcul de gradient magnitude et orientation
- 3) Détection des maxima locaux avec seuillage par hystérésis.
- 4) Chaînage des points
- 5) Approximation polygonale : par aire, orientation, etc...

Lissage : $b_n(i,j) = b_n(i) * b_n(j)$ ou $n = 4 * 2$

échelle : = échelle

Dérivée :

$$m_i(i,j) = i = \overline{i} = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \quad m_j(i,j) = j = \overline{j} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$$

for n pair :

$$b_n(i,j) = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} * n/2$$

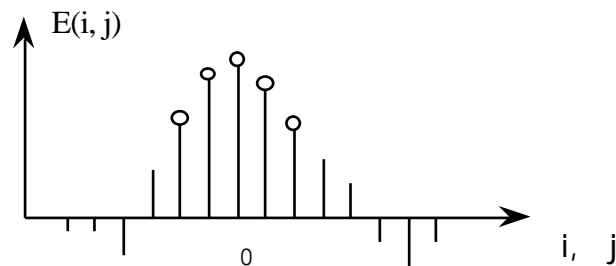
Détection des Pics dans le module du Gradient

Points de contraste : un extremum local en $E(i, j)$.

les points de contraste : $C(i, j)$
pour le gradient de la magnitude $E(i, j)$ et orientation (i, j)

Sur chaque point :

- 1) Déterminer le vecteur de la direction du maximum du gradient
- 2) Déterminer si le pixel $E(i, j)$ est un extremum dans la direction du gradient



$$C(i, j) = \begin{cases} 1 & E(i, j) > \text{Thr et} \\ & E(i-1, j-1) < E(i, j) < E(i+1, j+1) \\ 0 & \text{Sinon.} \end{cases}$$

ou $i = \text{Round}(\cos(\theta(i, j))) = \text{Round}(E_i(i, j)/E(i, j))$
et $j = \text{Round}(\sin(\theta(i, j))) = \text{Round}(E_j(i, j)/E(i, j))$

ensuite :

- Construire un graphe des "chaînes" des points de contraste
- Décrire les chaînes avec les segments et courbes.

Techniques:

- 1) Balayage des lignes et colonnes avec une extraction
- 2) Suivi des "crêtes" de contraste
- 3) La transformé de Hough

Transformée de Hough

(Duda et Hart, 1973)

Une technique "optimale" pour détecter les droites dans les images très bruitées.

Cette technique ne dépend pas de la continuité des droites.
Cependant, elle fournit des droites, pas des segments.

Une équation de droite s'exprime comme :

$$x \sin(\theta) - y \cos(\theta) + c = 0.$$

Pour chaque point (x, y) de l'image, il y a un ensemble de valeurs possibles pour les paramètres θ et c .

La transformée de Hough utilise un tableau $h(\theta, c)$ de "cellules" dans lequel on peut accumuler l'évidence pour les droites de contrastes.

Il faut discrétiser θ et c . Ceci dépend de la taille de l'image.

Par exemple :

Soit une image $N \times N$ pixels

Soit les valeurs entières d'angles $a = [0, 180]$

soit l'image de contraste $C(x,y)$.

$c_{\max} = 1.4N$

Allouez un tableau $h(\theta, c)$ for $1 \leq \theta < 180$ and $0 \leq c \leq 1.4N$

les valeurs entières de c de

pour $x = 0, N$

pour $y = 0, N$

pour $\theta = 1$ to 180

if $C(i,j) > 1$ then

{ $c = -x \cos(a) + y \sin(a)$

$h(\theta, c) = h(\theta, c) + 1$

}

Si des points de contrastes de l'image sont alignés, les droites correspondantes de l'espace de Hough passent toutes par un même point (θ, c) .

Un "pic" en $h(\theta, c)$ indique les valeurs θ et c d'une droite de contraste.

Généralisation de Transformée de HOUGH

Pour les cercles : une équation de cercle s'écrit :

$$(x - a)^2 + (y - b)^2 = r^2$$

On considère l'espace de Hough $h(a, b, r)$.

Chaque point (x, y) de l'image correspond à un cône de l'espace (a, b, r) .

Pour un rayon fixé, chaque point (x, y) correspond à un cercle de l'espace (a, b, r) .

Idée de l'algorithme :

Pour chaque rayon $r > 0$ on trace les cercles de l'espace de Hough correspondant aux points de l'image.

Lorsque tous les cercles se coupent en un même point, on a trouvé le bon rayon, les coordonnées (a, b) de ce point correspondent au centre du cercle.

Les operateurs Differentielles

Pour un signal en deux dimensions, les opérations différentielles sont le gradient et la Laplacien ² :

Le gradient est un vecteur :

$$\vec{\nabla} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix}$$

La Laplacien est une scalaire : $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$

Pour une fonction image $P(x,y)$:

La Gradient :

$$\nabla P(x,y) = \begin{pmatrix} \frac{\partial P(x,y)}{\partial x} \\ \frac{\partial P(x,y)}{\partial y} \end{pmatrix} = \begin{pmatrix} P_x(x,y) \\ P_y(x,y) \end{pmatrix}$$

La Laplacien

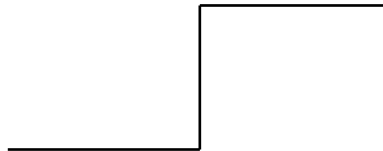
$$\nabla^2 P(x,y) = \frac{\partial^2 P(x,y)}{\partial x^2} + \frac{\partial^2 P(x,y)}{\partial y^2} = P_{xx}(x,y) + P_{yy}(x,y)$$

Les Deuxièmes dérivées

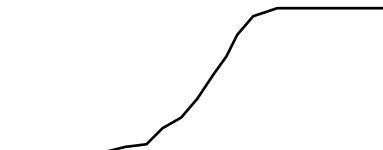
Il y a deux méthodes de détection des points de contraste :

- 1) les extrêmes de la dérivée première,
- 2) les passages à zéro de la dérivée seconde.

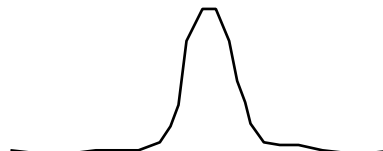
Contraste :



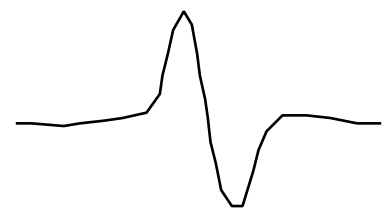
Lissage :



Dérivée 1^{ère}



Dérivée 2^{ème}



La deuxième différence peut être calculée par un filtre an-isotropique

$$\text{Laplacien: } \Delta^2 p(i,j) = \frac{\Delta^2 p(i,j)}{i^2} + \frac{\Delta^2 p(i,j)}{j^2}$$

$$i^2 = [1 \ -1] * [1 \ -1] = [-1 \ 2 \ -1]$$

$$i^2 = \begin{bmatrix} -1 & 2 & -1 \end{bmatrix} \quad j^2 = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix}$$

$$\text{Laplacien: } \Delta^2 p(i,j) = i^2 * p(i,j) + j^2 * p(i,j)$$

Il existe aussi des approximations discrètes du Laplacien .

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} + \begin{bmatrix} -1 & 2 & -1 \end{bmatrix}$$

Donc $L(u,v) = 4 - 2\cos(u) - 2\cos(v)$

le meilleur est :

$$\begin{bmatrix} -1 & 0 & -1 \\ 0 & 4 & 0 \\ -1 & 0 & -1 \end{bmatrix} + 2 \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} -1 & -2 & -1 \\ -2 & 12 & -2 \\ -1 & -2 & -1 \end{bmatrix}$$

Question :

Pourquoi le gradient exige deux convolutions,
et le Laplacien peut être calculé avec une seule convolution?

$$\text{Gradient : } \quad \| \quad p(i, j) \| = \sqrt{\left(\frac{\partial p(i,j)}{\partial i}\right)^2 + \left(\frac{\partial p(i,j)}{\partial j}\right)^2}$$

$$\text{Laplacien : } \quad \Delta p(i,j) = \frac{\partial^2 p(i,j)}{\partial i^2} + \frac{\partial^2 p(i,j)}{\partial j^2}$$

Passages à zéro dans la dérivée seconde

Les extrêmes de dérivée première correspondent à des zéros de la dérivée seconde

Intérêt : En théorie :

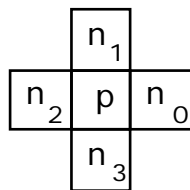
- 1) les passages à zéro donnent les contours fermés
- 2) Le passage à zéro peut être utilisé pour localiser les "points de contraste" avec une précision de moins d'un pixel (par interpolation).

En réalité :

Les passages à zéro détectent aussi beaucoup de petites régions non stables.

Test du passage à zéro:

Détection par 4 Voisins :



$$C(i, j) = \begin{cases} 1 & \text{si } (\text{signe}(n_0) \quad \text{signe}(n_2)) \text{ et } |n_0 - n_2| > 0 \\ & \text{ou } (\text{signe}(n_1) \quad \text{signe}(n_3)) \text{ et } |n_1 - n_3| > 0 \\ 0 & \text{Sinon} \end{cases}$$

Rend la détection plus stable :

$$C(i, j) = \begin{cases} 1 & \text{si } (\text{signe}(n_0) \quad \text{signe}(n_2)) \\ & \text{et } |n_0 - n_2| > 0 \\ & \text{et } |n_0 - n_2| > \text{seuil} \\ & \text{ou } (\text{signe}(n_1) \quad \text{signe}(n_3)) \\ & \text{et } |n_1 - n_3| > 0 \\ & \text{et } |n_1 - n_3| > \text{seuil} \\ 0 & \text{Sinon} \end{cases}$$

Inconvénient : les contours sont plus fermés.

Les operateurs Differentielles

Pour un signal en deux dimensions, les opérations différentielles sont le gradient et la Laplacien ² :

Le gradient est un vecteur :

$$\vec{\nabla} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix}$$

La Laplacien est une scalaire :

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

Pour une fonction $s(x,y)$:

La Gradient :

$$\nabla s(x,y) = \begin{pmatrix} \frac{\partial s(x,y)}{\partial x} \\ \frac{\partial s(x,y)}{\partial y} \end{pmatrix}$$

La Laplacien

$$\Delta^2 s(x,y) = \frac{\partial^2 s(x,y)}{\partial x^2} + \frac{\partial^2 s(x,y)}{\partial y^2}$$

si $s(x,y)$ est filtré par $G(x,y; \cdot)$: $s * G(x,y; \cdot) = \int_{s=-}^{t=-} s(x,y) G(x-s, y-t)$

La Gradient :

$$\nabla (s * G(x,y; \cdot)) = s * \nabla G(x,y; \cdot) = \begin{pmatrix} s * \frac{\partial G(x,y; \cdot)}{\partial x} \\ s * \frac{\partial G(x,y; \cdot)}{\partial y} \end{pmatrix}$$

La Laplacien

$$\Delta^2 (s * G(x,y; \cdot)) = s * \Delta^2 G(x,y; \cdot) = s * \left(\frac{\partial^2 G(x,y; \cdot)}{\partial x^2} + \frac{\partial^2 G(x,y; \cdot)}{\partial y^2} \right)$$

La fonction Gaussien

La fonction Gaussien est $G(x, y) = e^{-\frac{1}{2} \frac{x^2+y^2}{\sigma^2}}$

La fonction Gaussienne est invariante à la transformation affine:

$$T_a\{G(x, y)\} = G(T_a\{x\}, T_a\{y\})$$

Rappel en séance 2 on a vu que $x_r = x_c \frac{F}{z_c}$.

Donc la "taille" d'un objet est en proportion de $s = \frac{F}{z_c}$.

La taille (ou échelle) est une paramètre de la transformation affine.

La fonction Gaussienne est invariante à la transformation d'échelle :

$$T_s\{G(x, y)\} = G(T_s\{x\}, T_s\{y\}) = G(sx, sy)$$

Si on divise z_c (distance entre la caméra et l'objet) par deux, on double la taille.

$$G(x, y) = G(2x, 2y)$$

En 2-D, la Gaussienne symétrique circulaire est $G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$ avec $A = \frac{1}{2\sigma^2}$

La Gaussien est la fonction unique qui est symétrique circulaire et séparable.

On note que $G(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}} = e^{-\frac{x^2}{2\sigma^2}} * e^{-\frac{y^2}{2\sigma^2}}$

Ceci nous offre beaucoup d'intérêt pour la vitesse de calcul.

Les dérivées de la Gaussien

$$G(x, y) = e^{-\frac{1}{2} \frac{x^2+y^2}{\sigma^2}}$$

$$G_x(x, y) = -\frac{x}{\sigma^2} G(x, y)$$

$$G_{xx}(x, y) = \frac{x^2 - \sigma^2}{\sigma^4} G(x, y)$$

$$G_{xxx}(x, y) = -\frac{x^3 - 3x\sigma^2}{\sigma^6} G(x, y)$$

Les Filtres Numérique Gaussien

Parce que l'image est un signal échantillonné, il faut échantillonner $G(x, y)$

On obtient les filtres numériques par un simple échantillonnage de la fonction Gaussienne sur un intervalle $[-R, R]$.

On remplace x par nT_e , ou T_e est un

$$G(n) = G(nT_e) = e^{-\frac{1}{2} \frac{nT_e^2}{2}}$$

T_e est la pas d'échantillonnage.

Par convention l'on considère $T_e = 1$.

Donc, la forme numérique est $G(n) = e^{-\frac{1}{2} \frac{n^2}{2}}$

$$A = \int_{n=-} e^{-\frac{1}{2} \frac{n^2}{2}} dx \sqrt{2}$$

Il y a deux facteurs à maîtriser :

- La taille de la "support" $N = 2R+1$.
- la ratio $1/T_e$

Pour a:

Pour $N = 7$, les "ondes" de $W_R(f)$ dominent le spectre.

Pour $N = 9$, les ondes, on peut d'effet.

Pour b:

Il vaut mieux que $1/T_e = 1$

Les dérivées de la Gaussienne numérique sont :

$$G_x(n) = -\frac{n}{2} G(n)$$

$$G_{xx}(n) = \frac{n^2 - 2}{4} G(n)$$

$$G_{xxx}(n) = -\frac{n^3 - n}{6} G(n)$$

Pour la Gaussien en 2D. $G_x(i, j, \sigma)$

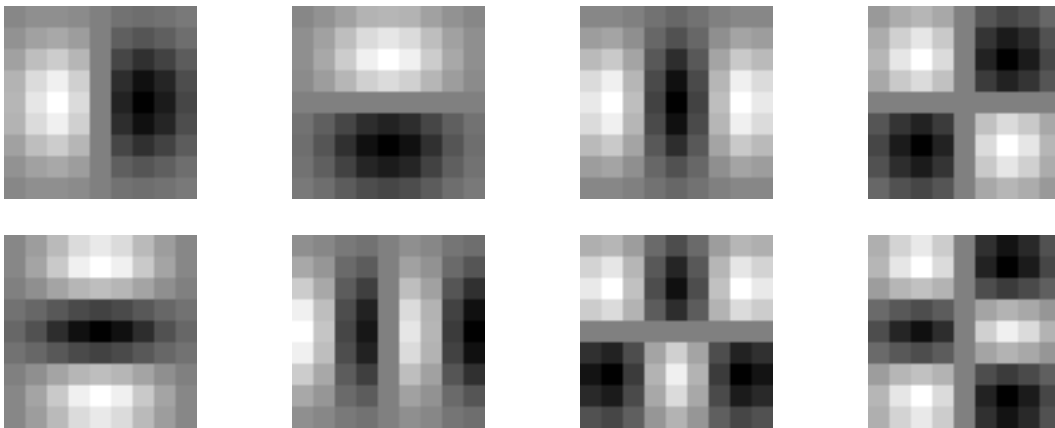
$$G_x(i, j, \sigma) = -\frac{i}{2\sigma^2} G(i, j, \sigma) = -\frac{i}{2\sigma^2} G(i, \sigma) * G(j, \sigma)$$

$$G_{xy}(i, j, \sigma) = \frac{ij}{4\sigma^4} G(i, j, \sigma) = -\frac{i}{2\sigma^2} G(i, \sigma) * -\frac{j}{2\sigma^2} G(j, \sigma)$$

Un vecteur de champs réceptifs forme une base de Taylor

$$G_a = (G_x, G_y, G_{xx}, G_{xy}, G_{yy}, G_{xxx}, G_{xxy}, G_{xyy}, G_{yyy})$$

Ceci donne la famille de champs réceptifs Gaussien



Les champs réceptifs Gaussien $G_x, G_y, G_{xx}, G_{xy}, G_{yy}, G_{xxx}, G_{xxy}, G_{xyy}, G_{yyy}$.

Note qu'il y a une paramètre σ . Ceci est la paramètre d'echelle. Ce détermine la limite de la résolution d'une description.

Les Dérivées de l'Image

Pour l'image $p(m, n)$, $p(m,n)$ est calculé par $G(m, n,) * p(m,n)$.

$$\text{ou } G(i,j) = \begin{pmatrix} G_x(i, j,) \\ G_y(i, j,) \end{pmatrix} \quad \text{et} \quad G(i, j,) = \begin{pmatrix} G_x(i, j,) \\ G_y(i, j,) \end{pmatrix}$$

$$\text{Gradient: } P(i,j) = \begin{pmatrix} \frac{P(i,j)}{x} \\ \frac{P(i,j)}{y} \end{pmatrix} = \begin{pmatrix} P * G_x(i, j,) \\ P * G_y(i, j,) \end{pmatrix} = \begin{pmatrix} P_x(i, j) \\ P_y(i, j) \end{pmatrix}$$

Il faut spécifier $P(i, j)$ pour calculer $P(i, j)$

$$\begin{aligned} \text{Laplacien: } \Delta P(i, j) &= \frac{\partial^2 P(i, j)}{\partial x^2} + \frac{\partial^2 P(i, j)}{\partial y^2} \\ &= P * G_{xx}(i, j,) + P * G_{yy}(i, j,) \\ &= P_{xx}(i, j) + P_{yy}(i, j) \end{aligned}$$

(Le Laplacien est une scalaire.)