

Détection de contours actifs en imagerie médicale

Dans une image en noir et blanc, un contour est caractérisé par une forte variation de luminosité. Discerner les contours d'une image permet d'extraire les principaux composants d'une image.

La détection de contours est un outil aux domaines d'applications variés : son utilisation dans l'imagerie médicale en terme de suivi et de diagnostic constitue un enjeu sociétal.

Parmi les différentes méthodes de détection de contours, il existe des algorithmes dynamiques : la courbe est initialisée et se déforme pour se rapprocher des contours de l'image. La méthode des contours actifs, également appelés *snakes*, a été développée en 1988 par Kass, Witkin and Terzopoulos [5] est une méthode dynamique. Similairement à la physique, chaque courbe se voit attribuée une quantité numérique d'énergie. La stabilité de la courbe est caractérisée par un minimum de cette énergie.

Dans le cadre des TIPEs, nous avons cherché à comprendre la théorie des contours actifs, que nous avons implémenté à l'aide de Python sur des images d'IRM et finalement, nous avons cherché à améliorer le rendu des courbes en les représentant par des B-Splines.

1 Snakes

On s'intéresse à des images semblables, afin de paramétrer de manière similaire la convergence du contour. Les contours envisagés sont simples, sans recoupement et lisses.

On représente l'image par une fonction $I : \mathbb{R}^2 \rightarrow [0, 1]$ et la courbe par une fonction $v : [0, 1] \rightarrow \mathbb{R}^2$ et l'im

1.1 Les contours actifs

Le principe de la recherche des contours actifs [1][2] consiste à calculer une suite de courbe qui converge vers un minimum d'énergie.

L'énergie de la courbe se décompose en deux termes :

- une énergie potentielle qui relève des données de l'image et quantifie la proximité de la courbe aux points à fort gradient :

$$E_{ext} = -\lambda(s)|\nabla I(v(s))|^2$$

- et une énergie interne, qui caractérise la forme de la courbe. Le coefficient α détermine son élasticité et le coefficient β sa rigidité:

$$E_{ext} = \alpha(s) (v(s))^2 + \beta(s) ([2]vs)$$

On obtient une valeur pour la courbe qui se calcule par :

$$E_{snakes}^* = \int_0^1 E_{int}(v(s)) + E_{ext}(v(s))ds \quad (1)$$

Il faut donc faire converger la courbe de manière à atteindre un minimum de E_{snakes} .

1.2 Equation d'Euler Lagrange

Pour trouver ce minimum local, la méthode des contours actifs repose sur l'équation d'Euler-Lagrange [4].

En physique, les équations de Lagrange sont des équations du mouvement qui permettent d'établir la trajectoire d'un objet avec des conditions initiale et finale fixées.

Le Lagrangien est défini par $L(q, \dot{q}, t) := T(q, \dot{q}, t) - V(q, t)$ où le terme $T(q, \dot{q}, t)$ est l'énergie cinétique totale du système et $V(q, t)$ est l'énergie potentielle.

Et la trajectoire empruntée par le système physique est celle qui minimise l'action $S(q(t)) = \int_{t_i}^{t_f} L(q, \dot{q}, t)$

Dans ce cas, le Lagrangien vérifie nécessairement l'équation d'Euler-Lagrange:

$$\frac{\partial L}{\partial q} - \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) = 0 \quad (2)$$

Il est possible de généraliser le Lagrangien aux ordres supérieurs $F = F(x, f(x), \dots, f^{(n)})$. On obtient alors :

$$\sum_{i=0}^n (-1)^i \frac{d^i}{dx^i} \left(\frac{\partial F}{\partial f^{(i)}} \right) = 0 \quad (3)$$

1.3 Application aux snakes

En prenant pour action l'énergie attribuée au snake (1), l'équation d'Euler Lagrange permet d'établir l'équation différentielle :

$$-\alpha v''(s) + \beta v^{(4)}(s) + \frac{\partial E_{ext}}{\partial v} = 0 \quad (4)$$

Le système d'équation établi afin d'assurer la convergence de la courbe vers une nullité du terme droite est alors

$$\alpha \frac{\partial^2 x}{\partial s^2} - \beta \frac{\partial^4 x}{\partial s^4} - \frac{\partial E_{ext}}{\partial x} = \frac{\partial x}{\partial t} \quad (5a)$$

$$\alpha \frac{\partial^2 y}{\partial s^2} - \beta \frac{\partial^4 y}{\partial s^4} - \frac{\partial E_{ext}}{\partial y} = \frac{\partial y}{\partial t} \quad (5b)$$

1.3.1 Résolution de l'équation

La courbe est représenté par une nombre fini de points. Les dérivées temporelles et spatiales peuvent être approximées par différences finies.

$$\frac{\partial x}{\partial s} \approx \frac{x_{(i+1,t_j)} - x_{(i,t_j)}}{\partial s} \quad (6)$$

Par itération, on obtient des expressions de u'' et de $u^{(4)}$

Pour les dérivées temporelles, on obtient également $\frac{\partial y}{\partial t} = -\tau(x_t - x_{t-1})$ avec τ le pas temporel.

Les équations (5a) et (5b) donnent les équations matricielles suivantes :

$$Ax_t + f_x(x_{t-1}, y_{t-1}) = -\tau(x_t - x_{t-1}) \quad (7a)$$

$$Ay_t + f_y(x_{t-1}, y_{t-1}) = -\tau(y_t - y_{t-1}) \quad (7b)$$

A est la matrice qui représente les dérivées par approximation des différences finies. C'est une matrice circulante dû au contour fermé étudié.

$$A = \begin{bmatrix} 2\alpha + 6\beta & -\alpha - 4\beta & \beta & 0 & \dots & \beta & -\alpha - 4\beta \\ -\alpha - 4\beta & 2\alpha + 6\beta & -\alpha - 4\beta & \beta & \dots & \dots & \beta \\ \beta & -\alpha - 4\beta & 2\alpha + 6\beta & -\alpha - 4\beta & \dots & \dots & \dots \\ 0 & \beta & -\alpha - 4\beta & 2\alpha + 6\beta & \dots & \dots & \dots \\ 0 & 0 & \beta & -\alpha - 4\beta & \dots & \dots & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

$$f_x(x, y) = \frac{\partial E_{ext}(x, y)}{\partial x} \text{ et } f_y(x, y) = \frac{\partial E_{ext}(x, y)}{\partial y}$$

Les fonctions $f_x(x, y)$ et $f_y(x, y)$ sont considérés constantes pendant le pas temporel. Elles sont alors évalués pour (x_{t-1}, y_{t-1}) .

En regroupant les termes définis pour t et $t - 1$, on obtient la relation itérative suivante :

$$x^t = (Id + \tau A)^{-1}(x^{t-1}) + \tau F(x^{t-1}, y^{t-1}) \quad (8a)$$

$$y^t = (Id + \tau A)^{-1}(y^{t-1}) + \tau F(x^{t-1}, y^{t-1}) \quad (8b)$$

1.4 Implémentation

1.4.1 Traitement de l'image

1. **Tons de gris** : L'image est convertie en tons de gris et le contraste est augmenté.
2. **Flou gaussien** : L'image est légèrement floutée. Le filtre gaussien est un filtre de lissage qui permet de réduire le bruit. Ainsi, la courbe ne va pas s'accrocher aux variations négligeables de l'image.
3. **Calcul du gradient** : Le gradient est caractéristique de la variation de la luminosité. Il est calculé selon les deux axes O_x et O_y . La valeur utilisé dans le calcul de l'énergie E_{ext} est le module du gradient $G = \sqrt{G_x^2 + G_y^2}$.



(a) Gradient selon O_x



(b) Gradient selon O_y



(c) Module du gradient

1.4.2 Initialisation

La matrice A et les fonctions $f_x(x, y)$ et $f_y(x, y)$ sont définies. La courbe est initialisée à sa valeur de départ.

1.5 Itération

Les valeurs de courbe sont calculés grâce aux formules (8).

2 Représentation des courbes

On souhaite implanter la détection de contour active dans un logiciel médical. Pour avoir un rendu visuel plus lisse, nous avons cherché une manière de représenter des courbes plus régulières. Les courbes retenues sont les B-Splines. Les B-Splines sont des généralisations des courbes de Bézier, qui permettent de paramétrer un tronçon de courbe par 4 points : ses points aux extrémités et leurs tangentes. Les B-Splines sont plus adaptés pour les courbes complexes, elles restent proches des points de contrôles et sont lisses.

2.1 B-Splines

Une B-Spline est décrite par des points de contrôles et par des noeuds t_0, \dots, t_n . [3] Elle se calcule comme une combinaison linéaire de ces points de contrôles:

$$S(t) = \sum_{i=0}^n B_{i,p}(t) P_i$$

On prends les noeuds égaux à $0, \dots, n$. Les coefficients $B_{i,p}(t)$ sont définis par récurrence selon l'**algorithme de De Boor**.

$$B_{i,0}(t) = \begin{cases} 1 & \text{si } t_i \leq t < t_{i+1} \\ 0 & \text{sinon} \end{cases}$$

$$B_{i,k+1}(t) := \frac{t - t_i}{t_{i+k+1} - t_i} B_{i,k}(t) + \frac{t_{i+k+2} - t}{t_{i+k+2} - t_{i+1}} B_{i+1,k}(t)$$

Comme nous avons choisis des noeuds équidistants, les B-Splines considérées sont uniformes, chaque point de contrôle à le même poids.

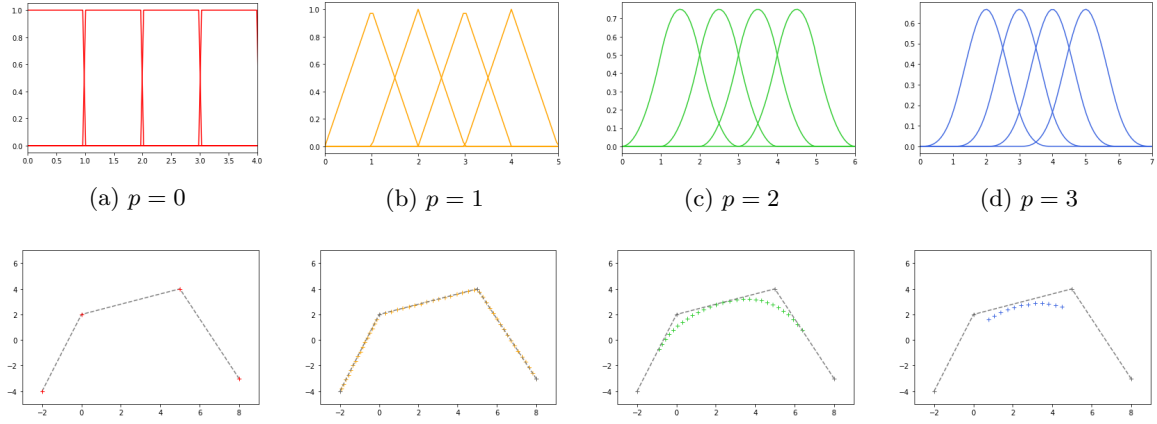


Figure 3: Représentation des $B_{i,p}$ et des B-Splines associées pour 4 points

La nullité des coefficients d'un point sur les tronçons éloignés, que l'on observe sur les tracés des coefficients, permet d'avoir une courbe locale. La modification d'un point de contrôle impacte seulement la partie de la courbe à proximité.

3 Résultats expérimentaux

L'implémentation de la méthode des contours actifs en Python permet d'obtenir les courbes suivantes.

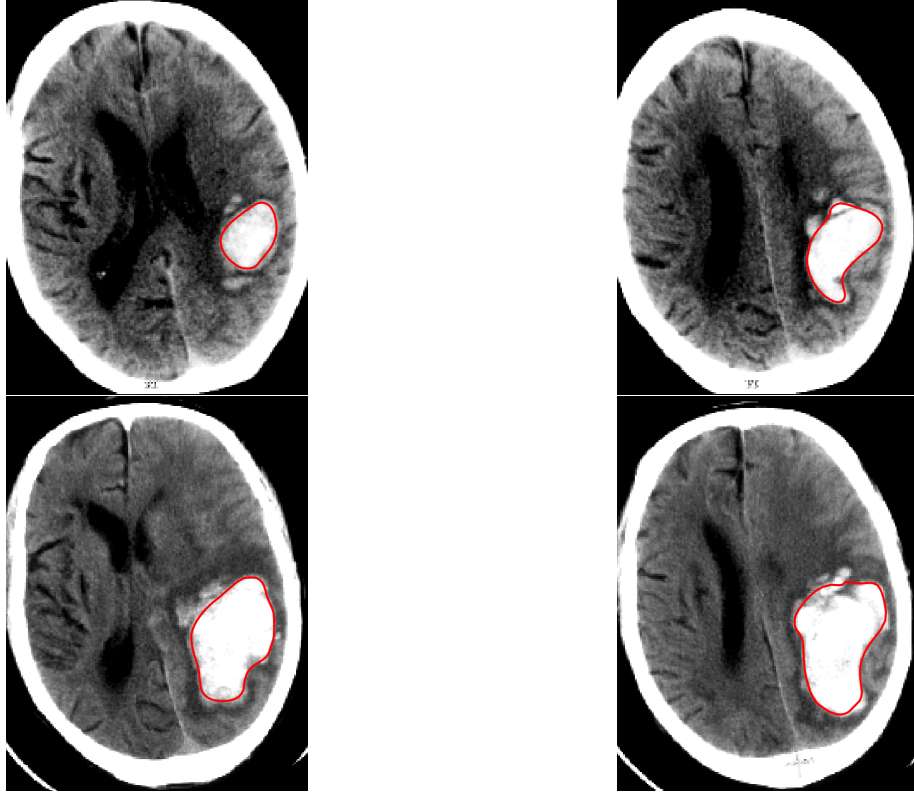


Figure 4: Détection de contours sur des images d'évolution d'une hémorragie intracérébrale [6]

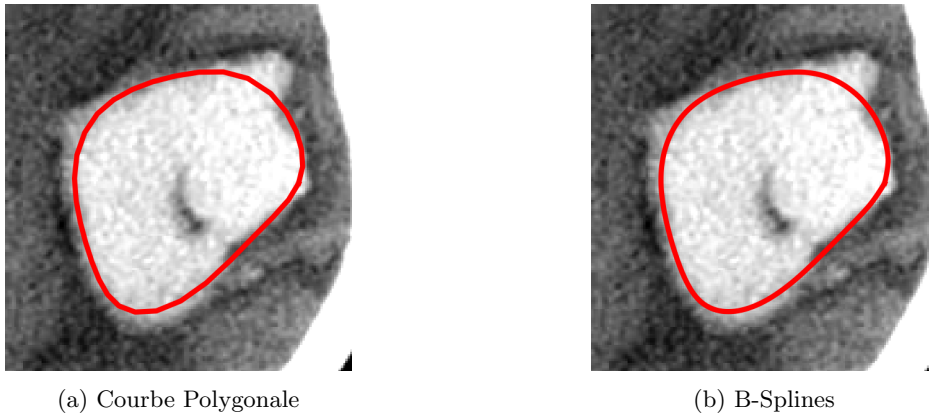


Figure 5: Comparaison des représentation de courbes

References

- [1] Marie-Odile Berger. “Les contours actifs : modélisation, comportement et convergence” ().
- [2] Muriel Gastaud. “Modèles de contours actifs pour la segmentation d’images et de vidéos” ().
- [3] Thomas Guilloid. “Interpolations, courbes de Bézier et B-Splines” ().
- [4] Floris Harmanni. “Higher Order Lagrangians” ().
- [5] Witkin Kass and Terzopoulos. *Snakes: Active Contour Models*. 1988.
- [6] S.M.Davis. “Medical Management of Haemorrhagic Stroke” ().