Traitement d'image : Segmentation par snakes

SINGARIN-SOLE — BELGUIDOUM

Avril 2024

Contents

1 Contexte

Dans ce TP, notre objectif est de détecter les contours d'une région, c'est-à-dire de détecter les zones de hautes fréquences sur une image. On peut, dans un premier temps, penser à calculer le gradient de notre image.

En effet, le gradient agit comme un filtre passe-haut et permet donc de détecter les hautes fréquences. Le gradient selon l'axe y va faire ressortir les contours horizontaux et le gradient selon l'axe x va faire ressortir les contours verticaux.

Cependant, le gradient est sensible au bruit et, pour limiter son influence, on convolue généralement l'image avec un noyau gaussien qui agit comme un filtre passe-bas. Ce noyau va donc lisser l'image et atténuer les variations brusques d'intensité afin de réduire le bruit. Néanmoins, on perd en résolution spatiale, donc cette méthode n'est pas la plus appropriée.

On implémente ainsi un détourage plus intelligent nommé ***Segmentation par snakes***.

2 Principe

Le principe de la méthode de segmentation par snakes est assez intuitif :

- 1. Effectuer un détourage grossier de l'objet
- 2. Ajuster cette forme automatiquement d'après les propriétés de l'image

L'objectif est de trouver la région globale où se situe l'objet dans l'image à l'aide d'un contour c. Le but va être de faire évoluer de manière itérative ce contour initial de sorte à le faire coller au contour de l'objet.

2.1 Définition du snake

On définit ainsi notre snake c comme une liste contenant les coordonnées de tous les points du contour : $\mathbf{c}(s) = \begin{bmatrix} x(s) \\ y(s) \end{bmatrix}, \quad s \in [0,1]$

La courbe c est donc une courbe lisse et fermée. Cette courbe prend en paramètre un réel entre 0 et 1 qui (abscisse curviligne). Etant fermée, on a évidemment $c(0) = c(1)$.

On réalise deux hypothèses sur c :

- c est une courbe lisse
- c se place sur les contours de la forme

2.2 Energie du snake

L'énergie externe du snake qui se definit comme suit :

$$E_{externe}\big(c(s)\big) = \int_0^1 E_{image}\big(c(s)\big) ds = \int_0^1 -\gamma \|\nabla\big(I\left(c(s)\right)\|^2\big) ds$$

Elle sert à guider le snake vers les contours significatifs de l'image en utilisant les informations du gradient d'intensité. C'est un composant essentiel qui permet au modèle de snake d'être efficace pour les tâches de segmentation d'image où les contours jouent un rôle crucial.

L'énergie interne, elle, se définit comme suit :

$$E_{interne}\big(c(s)\big) = \int_0^1 \left[\frac{\alpha}{2}\|c'(s)\|^2 + \frac{\beta}{2}\|c''(s)\|^2 ds\right.$$

Elle est totalement indépendante de l'image. Elle consiste en la somme de l'énergie élastique et de l'énergie de courbure. Ces énergies vont pénaliser respectivement la dérivée première et la dérivée seconde grâce aux paramètres alpha et beta.

Finalement, l'energie gobale du snake s'exprime ainsi :

$$E\big(c(s)\big) = E_{externe} + E_{interne} = \int_0^1 \left[\frac{\alpha}{2}\|c'(s)\|^2 + \frac{\beta}{2}\|c''(s)\|^2 - \gamma \|\nabla\big(I\left(c(s)\right)\|^2\big)\right] ds$$

On cherchera à minimiser cette valeur.

2.3 Minimisation de l'énergie

Le Snake optimale est celui qui va minimiser l'énergie globale. En posant ainsi une nouvelle variable F et en exprimant on obtient :

$$\begin{cases} \frac{\partial}{\partial s}\left(\frac{\partial F}{\partial c'}\right) - \frac{\partial^2}{\partial s^2}\left(\frac{\partial F}{\partial c''}\right) - \frac{\partial F}{\partial c} = 0 \\ F = \frac{\alpha}{2}\|c'(s)\|^2 + \frac{\beta}{2}\|c''(s)\|^2 - \gamma\|\nabla\big(I\left(c(s)\right)\|^2\big) \end{cases}$$

En identifiant, on obtient :

$$\begin{cases} \frac{\alpha}{2}\|c'(s)\|^2 + \frac{\beta}{2}\|c''(s)\|^2 - \gamma\|\nabla\big(I\left(c(s)\right)\|^2\big) = 0 \\ \alpha^{(2)}(s) - \beta^{(4)}(s) + \gamma\nabla\big[\|\nabla\big(I\left(c(s)\right)\|^2\big)\big] = 0 \end{cases}$$

C'est l'équation à résoudre de sorte que c'est le contour qui minimise E. Cependant cette équation différentielle est difficile à résoudre à cause du dernier membre. C'est pour cela qu'on a implémenter dans ce TP le schéma itératif de convergence.

On notera que selon l'algorithme qu'on va utiliser on va tomber sur des méthodes de résolutions bien connus.

En effet si on utilise dans l'algo une méthode de descente de gradient sur l'énergie on tombe sur le schéma de résolution de Euler explicite avec delta t qui joue le rôle du pas de la descente et si on utilise l'algorithme du proximal, on tombe sur le schéma de résolution de Euler implicite où Δ t joue aussi le rôle du pas.

3 Implémentation informatique

Pour expliquer, notre implémentation informatique, nous allons prendre l'exemple de l'image de la goutte. Commençons par calculer son gradient pour se donner une bonne idée concrète des contours :

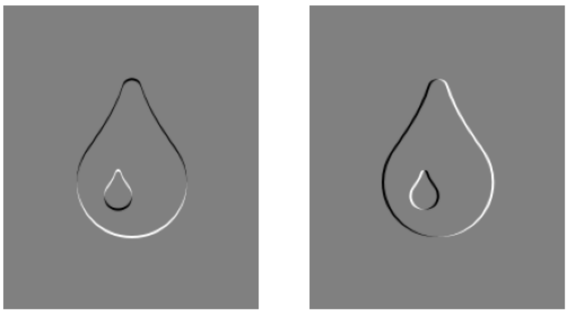

Figure 1: Gradient de l'image de la goutte en X et Y (lignes, colonnes

Dans notre cas on représente le Snake initial sous la forme d'un cercle. On se rappele que notre snake est discrétisé donc on pose K, le nombre de points. Ici pour K = 100 : Il faut faire attention à bien

Figure 2: Image d'une goutte avec son contour en vert

centrer notre cercle sur l'image pour obtenir un résultat cohérent.

Par la suite, on utilise l'algorithme de descente de gradient sur notre énergie global ce qui revient à utiliser la méthode de résolution d'Euler explicite : En posant : $A = I - \Delta t D$,

$$\begin{cases} x_{n+1} = A\Bigg(x_n + \Delta t \gamma \nabla_x \left[\|\nabla I\left[x_n, y_n\right)\right] \|^2 \right] \Bigg) \\ y_{n+1} = A\Bigg(y_n + \Delta t \gamma \nabla_y \left[\|\nabla I\left[x_n, y_n\right] \|^2 \right] \Bigg) \end{cases}$$

Ici, $D = \alpha D_2 - \beta D_4$ avec D_2 et D_4 les matrices de dérivée seconde et quatrième. Ces matrices sont implémentés dans le code en amont.

Ainsi, en choisissant les bon paramètres :

- $\alpha = 1$
- $\beta = 0.5$
- $\gamma = 15$
- $\Delta t = 0.1$

Et en effectuant 3 000 itérations, on obtient :

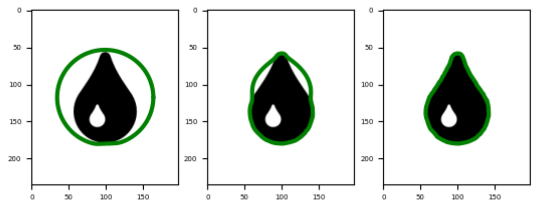

Figure 3: 1000 itérations, 2000 itérations, 3000 itérations

Pour déterminer les contours intérieures de l'image, on peut inverser les niveaux de gris de l'image d'origine et appliquer la méthode pat la suite. Ici, on obtient donc :

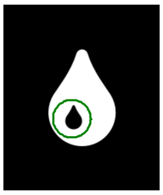

Figure 4: Nouveau contour de l'image de la goutte inversée

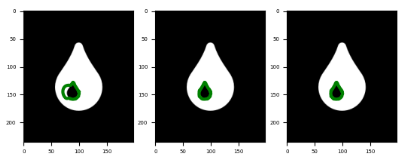

Figure 5: 1000 itérations, 2000 itérations, 3000 itérations

L'algorithme itératif permet d'ajuster la position du snake à l'image. À chaque itération, il calcule les gradients de l'image à la position des points du snake et ajuste ces points en fonction de l'attraction vers les zones à fort gradient (bords de l'objet) tout en maintenant la forme du snake via les matrices internes.

4 Expérimentation

Dans la pratique, la précision du détourage dépend grandement des propriétés de l'image choisie. Il est donc nécessaire d'utiliser les bon paramètres α, β et γ. Si on α, on pénalise l'énergie élastique, Si on β on pénalise l'énergie de courbure. En résumé, alpha et beta sont des régulateurs de la forme du snake : alpha agit comme une force qui tire le snake, le rendant plus droit et moins extensible, tandis que beta contrôle à quel point le snake peut se plier ou être fluide.

En ajustant ces paramètres, on peut affiner la segmentation pour qu'elle corresponde au mieux aux caractéristiques de l'objet à détecter dans l'image. On doit trouver un équilibre entre ces paramètres pour permettre au snake de suivre fidèlement les contours souhaités tout en ignorant le bruit et les détails non pertinents. De plus, afin d'améliorer nos résultats on pourra appliquer des opérations à notre image tel qu'un bruit par exemple.

Nous effectons ainsi une étude différente pour chaque image.

4.1 Exemple

On remarque qu'il y des cas où l'algorithme fonctionne mieux. Notamment sur les conteurs s'approchant d'un contour convexe.

Par exemple, pour l'image de feuille :

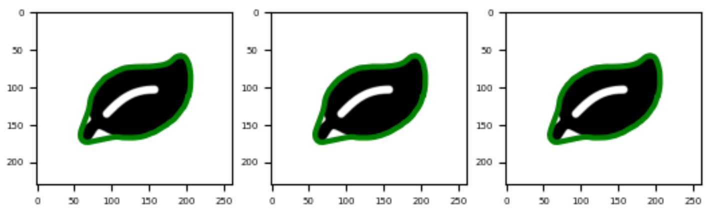

Figure 6: 2000 itérations, 4000 itérations, 6000 itérations

Et pour d'autres comme la pièce de puzzle où il y a beaucoup de fentes :

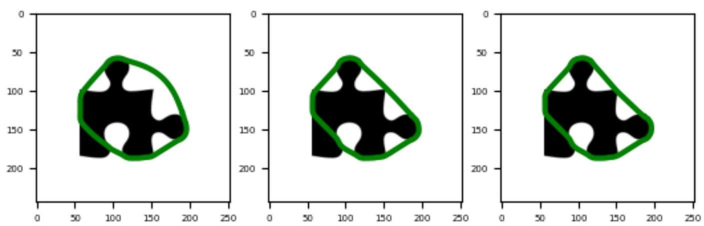

Figure 7: 2000 itérations, 4000 itérations, 6000 itérations

5 Conclusion

Nous avons employé l'algorithme de segmentation par snake pour identifier avec précision les contours dans les images. Grâce à une calibration minutieuse des paramètres alpha, beta, et gamma, nous avons renforcé la capacité du snake à épouser les contours véritables des objets.

Cette méthode nous a fourni une solution robuste et adaptable pour la segmentation d'image, essentielle pour les applications nécessitant une délimitation exacte des formes et des structures.