Voie d'Approfondissement HTI

IMA4509 | *Analyse des Contenus Visuels*

BE2

Restauration d'images par EDP

Nicolas Rougon

Institut Mines-Télécom / Télécom SudParis
Département ARTEMIS

nicolas.rougon@telecom-sudparis.eu

Ce BE a pour but de vous faire simuler et étudier expérimentalement divers modèles de filtrage par EDP d'images en niveaux de gris, associés à des espaces d'échelles conservatifs. Partant du modèle de Perona-Malik, vous envisagerez des modèles de complexité croissante obtenus par ajout de termes supplémentaires dans l'EDP génératrice initiale.

L'étude de chaque modèle se déroulera en 2 étapes :

1. **Implantation C/C++** d'une fonction générant la version filtrée à l'échelle t d'une image initiale. Vous vous appuierez pour cela sur le canevas logiciel fourni correspondant au filtre de Sobel (*cf.* `pdefilter.c`) :
   - les images seront représentées comme des tableaux 2D (*cf.* `globals.h utilities.c`);
   - les images en entrée/sortie seront au format PGM (*cf.* `utilities.c`);
   - les calculs seront effectués en *double*, le résultat final étant converti en *unsigned char* pour être sauvegardé.
2. **Etude expérimentale du modèle** sur des images convenablement sélectionnées dans la base disponible à l'adresse suivante :

   `www-public.tem-tsp.eu/~rougon/IMA4509/Labs/IMA4509-Lab3-Home.php`

   2.1. Vous décrirez d'abord l'influence de chaque hyperparamètre du modèle, en faisant varier sa valeur tout en conservant les autres constants. L'évaluation des résultats s'effectuera :

   (a) **subjectivement**, en appréciant visuellement l'image produite;

   (b) **objectivement**, en étudiant l'impact du filtrage sur la détection des contours de l'image. Vous utiliserez à cette fin les détecteurs de contours de la Toolbox Image Processing de Matlab[1] (*e.g.* `edge()`) ou d'OpenCV[2] (*e.g.* `Canny()`)

   2.2. Pour chaque image sélectionnée, vous chercherez ensuite à identifier un jeu de paramètres conduisant à une détection de contours optimale.

Le rendu, effectué par mail à `nicolas.rougon@telecom-sudparis.eu`, consistera en :

1. **un compte-rendu**, synthétisant vos conclusions et précisant, pour chaque image choisie, les valeurs des hyperparamètres utilisés (pour le modèle d'EDP et le détecteur de contours);
2. **le code C/C++ développé;**
3. **les images filtrées avec les jeux de paramètres optimaux.**

Dans la suite, les dérivées partielles seront notées de façon indicielle. Etant donnée une image en niveaux de gris u_0, on notera $u(\mathbf{x}, t)$ la version filtrée de u_0 à l'échelle t par l'EDP considérée, avec $u(\mathbf{x}, 0) = u_0(\mathbf{x})$ et des conditions de Neuman aux frontières de l'image.

[1] Disponible sur les serveurs Unix de TP dans le répertoire `/opt/matlab-disi-R2015a/bin`

[2] Les détecteurs utilisés dans le module IMA4103 sont disponibles sous `\\fuji\Etudiants_FUJI\IMA4509\OpenCV` dans les sous-répertoires `GradientEdgeDetection\Release` et `LaplacianEdgeDetection\Release`

1 Modèle de Perona-Malik

On considère tout d'abord le modèle de Perona-Malik défini par l'EDP génératrice :

$$u_t \; = \; \nabla \cdot \left(g\left(|\nabla u|\right) \nabla u \right) \tag{1}$$

g étant une fonction C^1-continue positive telle que : $g(0) = 1$ et $\lim\limits_{x \to +\infty} g(x) = 0$. On envisagera spécifiquement les fonctions de conduction suivantes :

- $g(x) = e^{-\left(\frac{x}{K}\right)^2}$
- $g(x) = \dfrac{1}{1 + \left(\frac{x}{K}\right)^2}$

$K > 0$ étant un seuil de contraste à spécifier.

L'EDP (1) est discrétisée par le schéma aux différences finies explicite suivant :

$$u_{i,j}^{(t+\delta_t)} \; = \; u_{i,j}^{(t)} \; + \; \delta_t \left[g_{i,j}^E \, u_{i+1,j}^{(t)} + g_{i,j}^W \, u_{i-1,j}^{(t)} + g_{i,j}^S \, u_{i,j+1}^{(t)} + g_{i,j}^N \, u_{i,j-1}^{(t)} - \sigma_{i,j} \, u_{i,j}^{(t)} \right] \tag{2}$$

avec :

$$\begin{aligned}
g_{i,j}^E &= g\left(u_{i+1,j}^{(t)} - u_{i,j}^{(t)} \right) \\
g_{i,j}^W &= g\left(u_{i-1,j}^{(t)} - u_{i,j}^{(t)} \right) \\
g_{i,j}^S &= g\left(u_{i,j+1}^{(t)} - u_{i,j}^{(t)} \right) \\
g_{i,j}^N &= g\left(u_{i,j-1}^{(t)} - u_{i,j}^{(t)} \right) \\
\sigma_{i,j} &= g_{i,j}^E + g_{i,j}^W + g_{i,j}^S + g_{i,j}^N
\end{aligned}$$

le pas de temps $\delta_t > 0$ du schéma étant contraint par la condition CFL : $\delta_t \leq \frac{1}{4}$.

1.1 Ecrivez une fonction permettant de calculer l'image $u(\mathbf{x}, t)$ à partir de u_0. Les arguments de cette fonction seront :

- le pas de temps δ_t,
- le nombre t d'itérations du schéma,
- le type de fonction de conduction g utilisée,
- le paramètre de contraste K.

1.2 En appliquant cette fonction sur des images convenablement sélectionnées, décrivez l'influence des différents paramètres du modèle selon la procédure indiquée en introduction, puis déterminez un jeu de paramètres autorisant une détection de contours optimale.

2 Modèle de Nordström

Le modèle de Nordström dérive du modèle de Perona-Malik par adjonction d'un terme d'attache à l'image originale. Son EDP génératrice est la suivante :

$$u_t \;=\; \nabla \cdot \left(g\left(|\nabla u|\right) \nabla u\right) \;+\; \lambda(u_0 - u) \tag{3}$$

où $\lambda > 0$ est un poids à spécifier.

2.1 A partir du code du modèle de Perona-Malik, écrivez une fonction calculant une version u d'une image u_0 filtrée par l'EDP (3). Ses arguments supplémentaires seront l'image originale u_0 et le poids λ du terme d'attache à u_0.

2.2 En appliquant cette fonction sur des images convenablement sélectionnées, décrivez l'influence du paramètre λ, puis déterminez un jeu de paramètres autorisant une détection de contours optimale.

3 Modèle de Nordström entropique

On obtient le modèle de Nordström entropique en augmentant le modèle de Nordström d'un terme de choc. Son EDP génératrice est la suivante :

$$u_t \;=\; \nabla \cdot \left(g\left(|\nabla u|\right) \nabla u\right) \;+\; \lambda(u_0 - u) \;-\; \mu |\nabla u| \, \mathsf{signe}(\Delta u) \tag{4}$$

$\mu \in \mathbb{R}^+$ un poids à spécifier. Le terme de choc est discrétisé via un schéma monotone conservatif. En posant :

$$\begin{aligned} D_x^+ u_{i,j} &= u_{i+1,j} - u_{i,j} & \qquad D_y^+ u_{i,j} &= u_{i,j+1} - u_{i,j} \\ D_x^- u_{i,j} &= u_{i,j} - u_{i-1,j} & \qquad D_y^- u_{i,j} &= u_{i,j} - u_{i,j-1} \end{aligned}$$

- Δu est discrétisé via un laplacien 4-connexe

$$\begin{aligned} \Delta u_{i,j} \;&\approx\; (u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) + (u_{i,j+1} - 2u_{i,j} + u_{i,j-1}) \\ &=\; D_x^+ u_{i,j} - D_x^- u_{i,j} + D_y^+ u_{i,j} - D_y^- u_{i,j} \end{aligned}$$

- si $\Delta u > 0$:

$$|\nabla u|_{i,j} \;\approx\; \sqrt{\min\left(D_x^+ u_{i,j}, 0\right)^2 + \max\left(D_x^- u_{i,j}, 0\right)^2 + \min\left(D_y^+ u_{i,j}, 0\right)^2 + \max\left(D_y^- u_{i,j}, 0\right)^2}$$

- si $\Delta u < 0$:

$$|\nabla u|_{i,j} \;\approx\; \sqrt{\max\left(D_x^+ u_{i,j}, 0\right)^2 + \min\left(D_x^- u_{i,j}, 0\right)^2 + \max\left(D_y^+ u_{i,j}, 0\right)^2 + \min\left(D_y^- u_{i,j}, 0\right)^2}$$

3.1 A partir du code du modèle de Nordström, écrivez une fonction calculant une version u d'une image u_0 filtrée par l'EDP (4). Son argument supplémentaire sera le poids μ du terme de choc.

3.2 En appliquant cette fonction sur des images convenablement choisies, décrivez l'influence du paramètre μ, puis déterminez un jeu de paramètres permettant une détection de contours optimale.