TP8 – Restauration d'images

L'*inpainting* (terme qu'on pourrait traduire en français par « retouchage ») est une technique permettant de reconstruire les régions endommagées d'une image. Deux cas de figure se présentent, selon que ces régions sont bien localisées ou non. L'application naturelle de cette technique consiste à retirer certains objets visibles sur une image, pour de bonnes raisons (restauration d'une image dégradée) ou de mauvaises raisons (élimination d'un personnage indésirable, par exemple). Dans ce TP, nous supposerons qu'aucune donnée supplémentaire n'est disponible (à l'inverse des « Eigenfaces » du TP2, ou d'autres méthodes fondées sur la texture). La restauration doit donc être uniquement guidée par les données non endommagées. Ceci peut être réalisé en adaptant les modèles variationnels de *débruitage*.

FIGURE 1 – Restauration d'image par *inpainting* : à gauche, image non dégradée ; au centre, image dégradée u_0 (la région dégradée D est indiquée en jaune) ; à droite, image u restaurée par *inpainting* (cf. exercice 3).

Exercice 1 : débruitage de type « Tikhonov »

La donnée est une image bruitée en niveaux de gris $u_0 : \Omega \to \mathbb{R}$ (on peut très facilement étendre la technique à des images couleur, cf. exercice 2-bis). L'objectif est de déterminer une nouvelle image u telle que :

- Les deux images u et u_0 soient « proches ».
- La nouvelle image u soit « lisse ».

Le modèle variationnel de débruitage de type « Tikhonov » consiste à minimiser l'énergie suivante :

$$E_{\text{Tikhonov}}(u) = \frac{1}{2} \iint_{\Omega} \left\{ \|\nabla u(x,y)\|^2 + \lambda \left[u(x,y) - u_0(x,y)\right]^2 \right\} dx\, dy \qquad (1)$$

où $\lambda > 0$ est un paramètre fixé par l'utilisateur.

Le minimiseur de (1) est solution de l'équation d'Euler-Lagrange suivante :

$$(\lambda I - \Delta)\, u(x,y) = \lambda\, u_0(x,y), \qquad \forall\, (x,y) \in \Omega \qquad (2)$$

où I est l'opérateur identité, et Δ l'opérateur laplacien. Après discrétisation par différences finies, (2) se réécrit :

$$\underbrace{\left[\lambda \mathbf{I}_N - \left(-\mathbf{D}_x{}^\top \mathbf{D}_x - \mathbf{D}_y{}^\top \mathbf{D}_y\right)\right]}_{\mathbf{A}} \mathbf{u} = \underbrace{\lambda\, \mathbf{u}_0}_{\mathbf{b}} \qquad (3)$$

où $\mathbf{u}$ et $\mathbf{u}_0$ sont des vecteurs de taille $N \times 1$ (N désigne le nombre de pixels) contenant les valeurs des niveaux de gris u et u_0, et $\mathbf{I}_N$ est la matrice identité (de taille $N \times N$). Enfin, $\mathbf{D}_x$ et $\mathbf{D}_y$ sont deux matrices bi-diagonales permettant d'approcher les composantes de ∇u par différences finies, et dont la définition a été vue en cours.

Complétez le script `exercice_1.m` de manière à implémenter le modèle de débruitage (1) par résolution de l'équation d'Euler-Lagrange discrète (3). Utilisez les fonctions `speye` pour construire la matrice identité, `spdiags` pour les matrices $\mathbf{D}_x$ et $\mathbf{D}_y$, et l'opérateur \ (*backslash*) pour la résolution du système. Étudiez notamment l'influence du paramètre λ, et la qualité de la préservation des contours.

Exercice 2 : débruitage par variation totale

La régularisation quadratique $\iint \|\nabla u(x,y)\|^2\,dx\,dy$ ne permet pas de restaurer les contours saillants. On lui préfère généralement la « variation totale » $\iint \|\nabla u(x,y)\|\,dx\,dy$, mais l'énergie à minimiser devient non différentiable. On peut toutefois utiliser l'approximation différentiable suivante :

$$E_{\mathrm{VT}}(u) = \frac{1}{2}\iint_\Omega \left\{\sqrt{\|\nabla u(x,y)\|^2+\epsilon} + \lambda\left[u(x,y)-u_0(x,y)\right]^2\right\} dx\,dy \tag{4}$$

où ϵ est un paramètre permettant de garantir la différentiabilité (on choisit $\epsilon = 0,01$).

L'équation d'Euler-Lagrange associée est non linéaire, mais elle peut être résolue itérativement par un schéma numérique de type « point fixe » :

$$\left[\lambda I - \mathrm{Div}\left(\frac{1}{\sqrt{\|\nabla u^{(k)}(x,y)\|^2+\epsilon}}\,\nabla\right)\right] u^{(k+1)}(x,y) = \lambda\, u_0(x,y), \qquad \forall\,(x,y)\in\Omega \tag{5}$$

Après discrétisation, l'itération (5) s'écrit :

$$\underbrace{\left[\lambda \mathbf{I}_N - \left(-{\mathbf{D}_x}^\top \mathbf{W}^{(k)}\mathbf{D}_x - {\mathbf{D}_y}^\top \mathbf{W}^{(k)}\mathbf{D}_y\right)\right]}_{\mathbf{A}^{(k)}} \mathbf{u}^{(k+1)} = \underbrace{\lambda\,\mathbf{u}_0}_{\mathbf{b}} \tag{6}$$

où $\mathbf{W}^{(k)}$ est la matrice diagonale, de taille $N\times N$, constituée des coefficients $\frac{1}{\sqrt{\|\nabla u^{(k)}(x,y)\|^2+\epsilon}}$, $(x,y)\in\Omega$.

Complétez le script `exercice_2.m` de manière à implémenter le modèle (4) de débruitage par variation totale, au moyen des itérations de point fixe (6). La qualité de la restauration des contours devrait être grandement améliorée. Faites ensuite une copie de ce script, de nom `exercice_2_bis.m`, que vous modifierez de façon à appliquer la méthode à des images RVB : vous pourrez, par exemple, appliquer le même algorithme indépendamment à chacun des trois canaux. Testez ce script sur l'image `lena.bmp`.

Exercice 3 : *inpainting*

La résolution de l'*inpainting* par variation totale consiste en une modification très simple du modèle de débruitage (4). On suppose dans ce cas que la donnée u_0 n'est pas fiable sur une région $D\subset\Omega$. Le terme d'attache aux données $\iint [u(x,y)-u_0(x,y)]^2\,dx\,dy$ ne doit donc être calculé que sur $\Omega\backslash D$. En l'absence de bruit, il est également inutile d'appliquer la régularisation en dehors de D. Le modèle devient donc :

$$E_{inpainting}(u) = \frac{1}{2}\iint_D \sqrt{\|\nabla u(x,y)\|^2+\epsilon}\;dx\,dy + \frac{1}{2}\iint_{\Omega\backslash D} \lambda\left[u(x,y)-u_0(x,y)\right]^2\;dx\,dy \tag{7}$$

ce qui conduit aux itérations de point fixe suivantes :

$$\underbrace{\left[\lambda\,\mathbf{W}_{\Omega\backslash D} - \mathbf{W}_D\left(-{\mathbf{D}_x}^\top \mathbf{W}^{(k)}\mathbf{D}_x - {\mathbf{D}_y}^\top \mathbf{W}^{(k)}\mathbf{D}_y\right)\right]}_{\mathbf{A}^{(k)}} \mathbf{u}^{(k+1)} = \underbrace{\lambda\,\mathbf{u}_0}_{\mathbf{b}} \tag{8}$$

où $\mathbf{W}_D$ est la matrice diagonale constituée des valeurs de la fonction caractéristique χ_D de D : $\chi_D(x,y)=1$ si $(x,y)\in D$, $\chi_D(x,y)=0$ sinon.

En vous inspirant du script `exercice_2_bis.m`, complétez le script `exercice_3.m` de manière à restaurer l'image de la fleur (cf. figure 1), pour laquelle le domaine D est fourni.

Dans le cas où l'on souhaite « effacer » un texte, comme par exemple un sous-titre, une autre solution consiste à déterminer D par segmentation. Faites une copie du script `exercice_3.m`, de nom `exercice_3_bis.m`, que vous modifierez de manière à effacer le texte de l'image `grenouille.png`. Pour cette image, il semble opportun d'effectuer la segmentation par classification, en utilisant la couleur comme critère (il est rappelé que le jaune n'est pas une couleur primaire, mais le mélange du rouge et du vert).