**Image–Signal–Simulation**

**Travaux pratiques**
**Feuille 3 :**
**Zoom d'images par minimisation de la variation totale**

Dans ce TP, nous allons considérer une méthode d'optimisation pour zoomer des images. Nous considèrerons un modèle d'échantillonnage simple. Étant donnée une image de départ $u$, de taille $KN \times KN$, l'image sous-échantillonnée, dans un rapport $K$, $v$ est définie par

$$v_{i,j} = \frac{1}{K^2} \sum_{k,l=0}^{K-1} u_{Ki+k,Kj+l},$$

pour $(i,j) \in \{0,\ldots,N-1\}^2$. On note l'opérateur d'échantillonnage ainsi défini $Q$. On a donc $v = Q(u)$.

On notera

$$\mathcal{C} = \{w \in I\!\!R^{(KN)^2}, Q(w) = v\}.$$

On sait donc que $u \in \mathcal{C}$, c'est la seule information dont nous sommes sûr concernant $u$.

Pour définir une méthode d'optimisation permettant de zoomer une image $u$, il est donc naturel de considérer le problème :

$$(P) : \begin{cases} \text{minimiser } E(w) \\ \text{sous la contrainte } w \in \mathcal{C}, \end{cases}$$

pour une énergie $E$ bien choisie.

Nous considèrerons dans cet exercice la minimisation d'une approximation différentiable de la variation totale. Pour cela, nous définissons

$$\nabla w_{i,j} = \begin{pmatrix} (D_x w)_{i,j} \\ (D_y w)_{i,j} \end{pmatrix} = \begin{pmatrix} w_{i+1,j} - w_{i,j} \\ w_{i,j+1} - w_{i,j} \end{pmatrix}$$

pour $(i,j) \in \{0,\ldots,KN-1\}^2$ (on supposera que $w$ est périodisée en dehors de son support).

Par ailleurs, nous approximerons le module du gradient, dans la variation totale, par

$$\varphi_\beta(|\nabla w_{i,j}|^2),$$

avec

$$\varphi_\beta(t) = \sqrt{t+\beta},$$

pour $\beta > 0$. En pratique, on prendra $\beta = 0.01$. L'approximation de la variation totale est alors définie par

$$E_\beta(w) = \sum_{i,j=0}^{KN-1} \varphi_\beta\left(|\nabla w_{i,j}|^2\right).$$

**Exercice 1.**

(1) Vérifier que $\mathcal{C}$ est un espace affine (i.e.: de la forme $u_0 + \mathcal{C}'$, pour un point $u_0$ et un espace vectoriel $\mathcal{C}'$). Décrire l'espace vectoriel $\mathcal{C}'$ définissant sa direction.

(2) Vérifier que l'opérateur qui à $w \in I\!\!R^{(KN)^2}$ associe $P_{\mathcal{C}}(w) \in I\!\!R^{(KN)^2}$, définie par

$$P_{\mathcal{C}}(w)_{i,j} = w_{i,j} - Q(w)_{[\frac{i}{K}],[\frac{j}{K}]} + v_{[\frac{i}{K}],[\frac{j}{K}]},$$

où $[t]$ représente la partie entière de $t$, est la projection orthogonale sur $\mathcal{C}$.

(3) Vérifier que tous les éléments de $\mathcal{C}$ ont la même moyenne et en déduire que $(P)$ a une solution. Proposer un exemple pour lequel $(P)$ a plusieurs solutions.

(4) On a vu en cours que

$$\nabla E(w) = 2\left(D_x^*(X) + D_y^*(Y)\right),$$

où $D_x^*$ et $D_y^*$ sont les même que dans le TP1, et

$$X_{i,j} = \varphi'_\beta(|\nabla w_{i,j}|^2) D_x w_{i,j},$$
$$Y_{i,j} = \varphi'_\beta(|\nabla w_{i,j}|^2) D_y w_{i,j},$$

pour $(i,j) \in \{0, \dots, KN-1\}^2$, et où $\varphi'_\beta(t)$ est la dérivée de $\varphi_\beta$ au point $t$.

(5) Détailler l'algorithme du gradient projeté pour résoudre $(P)$.

(6) Lire le programme *echantillonne_ image* et le programme *zoom_ TV*. Dire (dans le détail) ce que font les fonctions qui les composent. Si besoin, argumentez votre réponse par un calcul.

(7) Lancer les différentes cellules du programme python et commenter les résultats obtenus.

(8) Modifier le module *zoom_ TV* pour pouvoir utiliser un pas de Armijo.

(9) Faire tourner le module *zoom_ TV* pour plusieurs valeurs du pas constant et pour le pas de Armijo. Concluez sur le choix du nombre d'itérations et du pas.

**Exercice 2.**

(1) Ecrire un programme suivant l'algorithme du gradient à pas constant et le pas de Armijo, pour minimiser une fonctionnelle permettant de résoudre $(P)$ par pénalisation. Pour cela, vous considèrerez la fonctionnelle:

$$F(w) = E(w) + \lambda \sum_{i,j=0}^{N-1} (Q(w)_{i,j} - v_{i,j})^2,$$

pour une grande valeur de $\lambda$.

(2) Adapter le programme pour choisir le pas conduisant à la plus grande descente, vu en cours.

(3) Comparer les vitesses de convergence pour les différentes stratégies de choix de pas à votre disposition.

(4) Lorsque $Q$ est l'identité (i.e. $K = 1$), quelle est l'énergie minimisée par votre programme? Comparer les résultats qualitatifs à ceux du TP 1.

**Exercice 3.**

(1) Ecrire une fonction permettant de calculer l'opérateur de convolution

$$\begin{array}{rcl} H : \mathbb{R}^{N^2} & \longrightarrow & \mathbb{R}^{N^2} \\ w & \longmapsto & Hw \end{array}$$

avec, pour tout $(m,n) \in \{1, \dots, N\}^2$,

$$(Hw)_{m,n} = \sum_{m',n'=-1}^{1} w_{m+m',n+n'}$$

(Les images sont supposées périodisées.)

(2) Ecrire un programme suivant l'algorithme du gradient à pas constant et le pas de Armijo, pour minimiser une fonctionnelle permettant de résoudre

$$F(w) = TV_\varepsilon(w) + \lambda \|Hw - v\|^2,$$

pour des valeurs de $\varepsilon$ et $\lambda > 0$.

(3) Tester différentes valeurs de paramètre pour déconvoluer une image

$$v = Hu + b$$

où $u$ est une image idéale et $b$ est un bruit Gaussien d'écart type 2.