

MVA 2015-16 “Compressed Sensing” – TP4 – Lilian Besson

Veillez trouver en pièce jointe mon code pour le TP4¹ du cours “Compressed Sensing”.

Remarques et résultats

- Comme pour le TP4, j’ai tenté avec plusieurs images (`boat`, `lena` et `rubik3`). Les résultats de l’algorithme primal-dual pour l’impainting semblent similaires sur chaque image, c’est rassurant.
- Sur une image “moins complexe” comme ce carré, la “complétion par interpolation intelligente” (*inpainting*) fonctionne encore mieux (et c’est très naturel).
- On observe que, comme prévu, les énergies – les normes $\|f_k\|_{TV}$ des variations totales – diminuent au cours des étapes alternantes primales-duales.

Illustrations et résultats pour chaque exercice

Ces exemples² montrent les résultats obtenus pour les exercices 1, 2 et 3 (*primal-dual inpainting*) :

Exercice 1

Exemple d’impainting sur l’image “Lena” :



Figure 1 – (Ex1) The “Lena” image f and an artificially altered observation y (80% missing pixels!).



Figure 2 – (Ex1) Impainted image from “Lena”, after 300 steps of the primal-dual algorithm.

1. http://www.numerical-tours.com/matlab/inverse_5_inpainting_sparsity/.
2. Les autres graphiques sont tous dans le dossier `fig` fournit dans le fichier zip envoyé avec ce TP4, notamment pour Ex. 1 et 3.

On constate que l'énergie diminue au cours des étapes de l'algorithme :

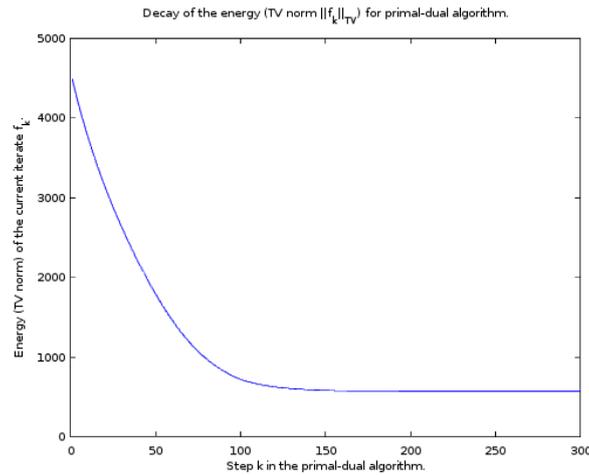


Figure 3 – (Ex1) Decay of TV norms during inpainting on “Lena” with the primal-dual algorithm.

Et, puisqu'on dispose de l'image initiale f , on peut aussi observer que le SNR entre f et f_k augmente au cours des étapes de l'algorithme :

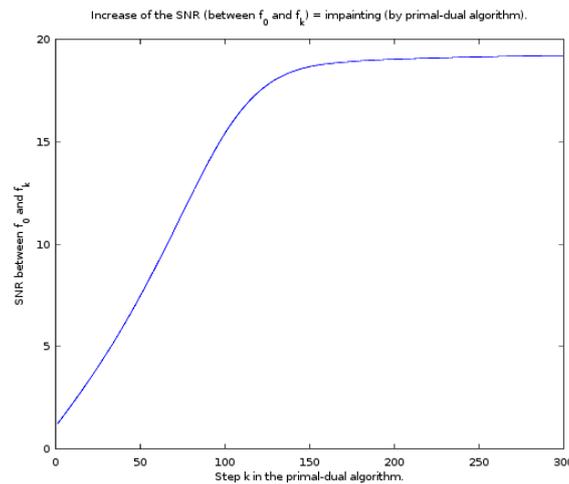


Figure 4 – (Ex1) Increase of SNR during inpainting on “Lena” with the primal-dual algorithm.

Exercice 2

Use the primal dual scheme to perform regularization in the presence of noise :

$$\min_{\|y - \Phi(f)\| \leq \varepsilon} \|\nabla f\|_1$$

J'ai d'abord généré une image bruitée de “Lena”, avec un bruit blanc Gaussien d'amplitude $\sigma = 0.1$. A gauche l'image f_0 initiale, puis l'image bruitée, puis l'image bruitée à laquelle on a enlevé 70% des pixels :



Figure 5 – (Ex2) The “Lena” image f and an artificially altered observation y (70% missing pixels and white noise $\sigma = 0.1$).

En appliquant le même algorithme, on obtient une image débruitée + complétée, qui me semble plutôt satisfaisante :

Final inpainted image with primal-dual algorithm (300 steps, TV norm = 1355.012, SNR = 16.753).



Figure 6 – (Ex2) Impainted image from noisy “Lena”, after 300 steps of the primal-dual algorithm.

On constate ici aussi que l’énergie diminue au cours des étapes de l’algorithme, et qu’elle commence avec une valeur bien plus élevée que dans le premier exemple de l’exercice 1 (parce que l’observation y est bruitée en plus d’avoir des pixels manquants) :

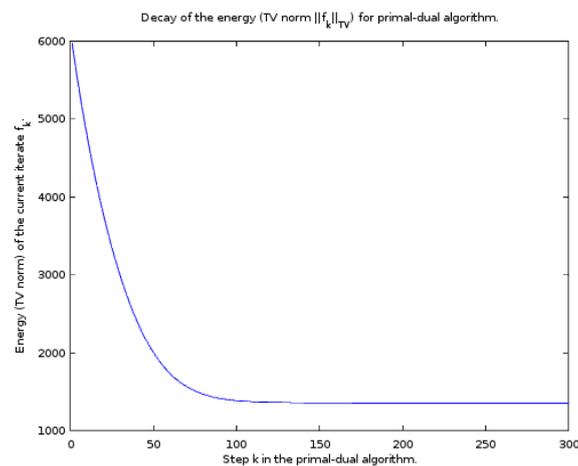


Figure 7 – (Ex2) Decay of TV norms during inpainting on “Lena” with the primal-dual algorithm.

Et, puisqu’on dispose de l’image initiale f , on peut encore observer que le SNR entre f et f_k augmente au cours des étapes de l’algorithme :

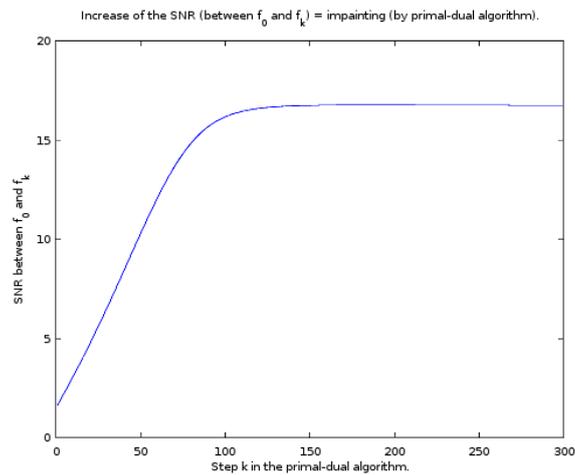


Figure 8 – (Ex2) Increase of Signal-to-Noise Ratio (SNR) during inpainting on noisy “Lena” with the primal-dual algorithm.

Note : je ne sais pas s’il fallait trouver un nouvel algorithme ou juste essayer d’appliquer le premier à une image bruitée. Je ne sais pas comment calculer l’opérateur proximal dans le cas d’un espace H qui n’est plus un hyperplan. Dans cet exercice, l’espace H n’est plus un sous-espace vectoriel (e.g. pas stable par somme).

J’ai (un peu) cherché par moi-même et dans quelques articles, sans trouver quelque chose de satisfaisant.

References :

- “Linear inverse problems with noise: primal and primal-dual splitting”, by F.-X. Dupé, J. Fadili and J.-L. Starck (PDF).
- “A first-order primal-dual algorithm for convex problems with applications to imaging”, by A. Chambolle and T. Pock (PDF).

Exercice 3

Première expérience Exemple d’impartialité sur le carré demandé par l’énoncé :



Figure 9 – (Ex3) The square image f and an artificially altered observation y .

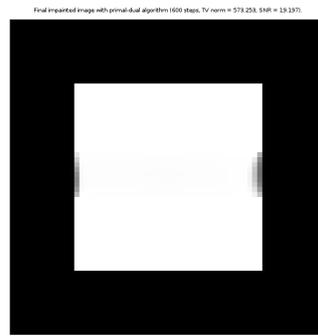
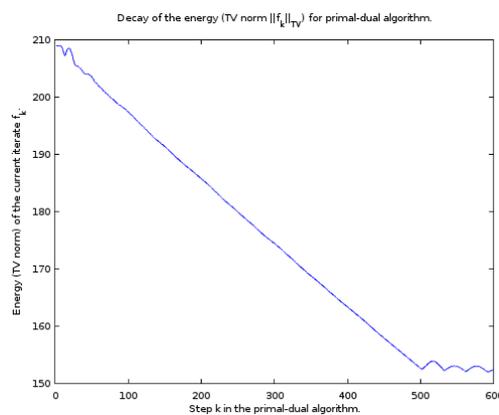


Figure 10 – (Ex3) Impainted square, after 600 steps of the primal-dual algorithm.

On constate encore une fois que l'énergie diminue au cours des étapes de l'algorithme :

Figure 11 – (Ex3) Decay of TV norms during inpainting on the square image with the primal-dual algorithm.

Et on peut aussi afficher l'image en train d'être complétée durant l'algorithme primal-dual (lire de gauche à droite puis de haut en bas) :

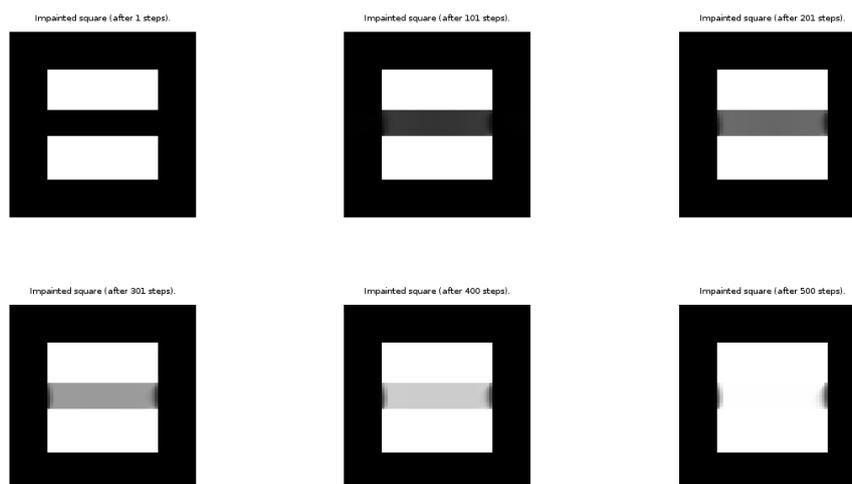


Figure 12 – (Ex3) Evolution of the inpainted square image during the primal-dual algorithm (500 steps).

Comme prévu, on constate que cette procédure de complétion par l'algorithme primal-dual fonctionne très bien ici, sur une image aussi "simple" que ce carré, la procédure fonctionne encore mieux que sur "Lena" ou d'autres images plus compliquées. Simple et compliqué dans ce contexte signifient surtout que les dommages sont grands et localisés (e.g. pour ce carré), ou petits et omniprésents (e.g. pour Lena où on avait enlevé 80% des pixels aléatoirement).

Deuxième expérience J’ai aussi voulu tester la même procédure sur un carré modifié différemment :



Figure 13 – (Ex3) The square image f and an artificially altered observation y_2 .

Je voulais observer que la partie grise devrait être complétée avant la partie noire, et bien plus rapidement, pour finalement obtenir une image presque parfaitement complétée, comme pour le premier exemple :

Final inpainted image with primal-dual algorithm (450 steps, TV norm = 152.867, SNR = 19.197).

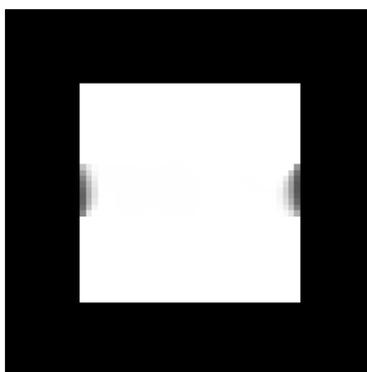


Figure 14 – (Ex3) Impainted square, after 450 steps of the primal-dual algorithm.

Ici aussi, on constate que l’énergie diminue au cours des étapes de l’algorithme. Il est intéressant de noter qu’il y a deux pentes successives, d’abord une forte descente puis une descente moins forte. Cela correspond à l’effacement de la bande grise verticale (effacement “facile” donc forte décroissance), puis de la bande noire horizontale (effacement plus “difficile” donc plus faible décroissance) :

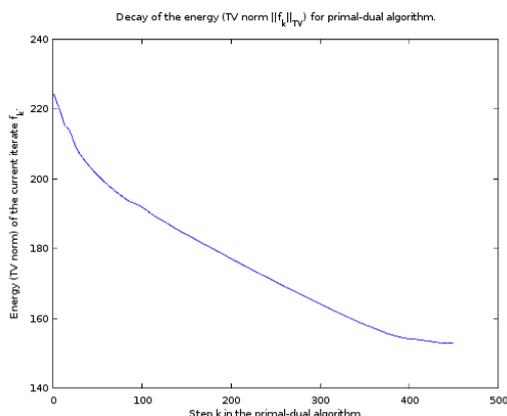


Figure 15 – (Ex3) Decay of TV norms during inpainting on the square image with the primal-dual algorithm.

On peut encore afficher l’image en train d’être complétée durant l’algorithme primal-dual, et comme je l’espérais on constate que la bande verticale (grise à 0.20% de blanc) est corrigée bien plus vite que la bande noire du premier exemple, et qu’elle est corrigée avant la bande noire horizontale. Une fois la bande grise corrigée, la bande noire s’efface progressivement :

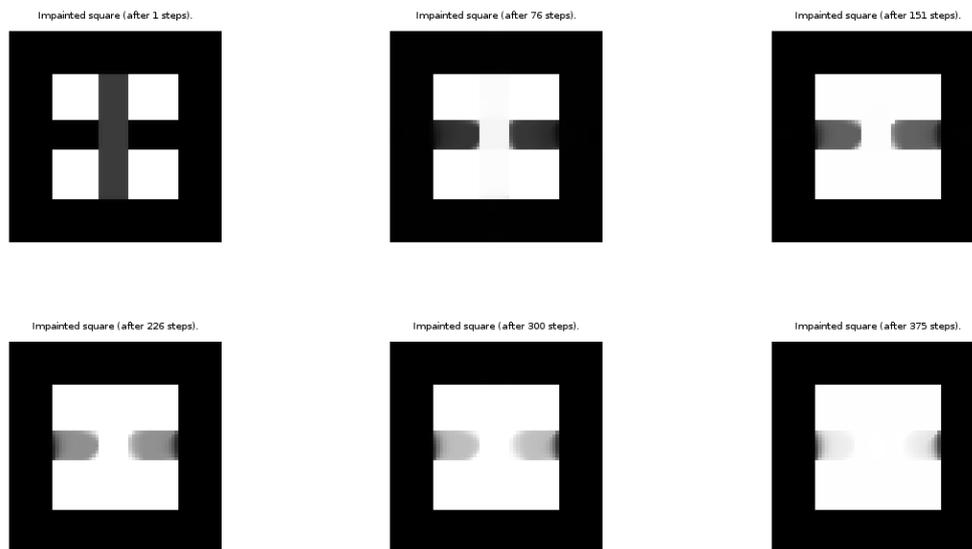


Figure 16 – (Ex3) Evolution of the inpainted square image during the primal-dual algorithm (450 steps).

Pour ce second exemple, la procédure de complétion fonctionne aussi très bien. Yay!

Sur des images en couleurs ? J'ai aussi un peu essayé d'appliquer cet algorithme à des images en couleurs ($\mathbb{R}^{3 \times N \times M}$), mais je n'ai pas réussi à obtenir des graphiques assez intéressants et jolis pour être inclus.

Je reste bien sûr à votre disposition si besoin.