IMA201 - TP01
Élève: OLIVEIRA MACHADO DE SOUSA, Lucas

Questions

2. Visualisation et utilisation de gimp

2.1. Zooms

- Que fait gimp pour afficher l'image en plus grand?

  Gimp fait une interpolation PPV (plus proche voisin).

- Ouvrez l'image **lena_peit.tif** et zoomez-la d'un facteur 2 également. Comparez le résultat par rapport au zoom de la petite image que vous avez produite. Quelle hypothèse pouvez-vous faire sur la génération de **lena_petit.tif**?

Probablement l'image lena_petit.tif est une transformation de l'image lena.tif fait avec interpolation. On voit que l'image réduite a moins de détails, on peut voir des gros pixels noirs sur le chapeau de Lena, etc., et on voit aussi que dans l'image lena_petit.tif il y a une petite tache noire au dessus de son sourcil que n'apparait pas dans l'image réduite sans interpolation.

Fig. 1 - Lena.tif réduit sans interpolation (à gauche) et Lena_petite.tif (à droite).

2.2. Espace couleurs

- Essayez de transformer les fleurs jaunes en fleurs bleues avec le bouton Hue. Comprenez-vous pourquoi les deux positions extrêmes de ce boutons font, en fait, la même transformation?

L'espace de couleurs HSL (Hue, Saturation, Lightness, ou Teinte, Saturation, Luminosité) est un cylindre où la dimension angulaire est la teinte (hue), comme on peut voir ci-dessous.

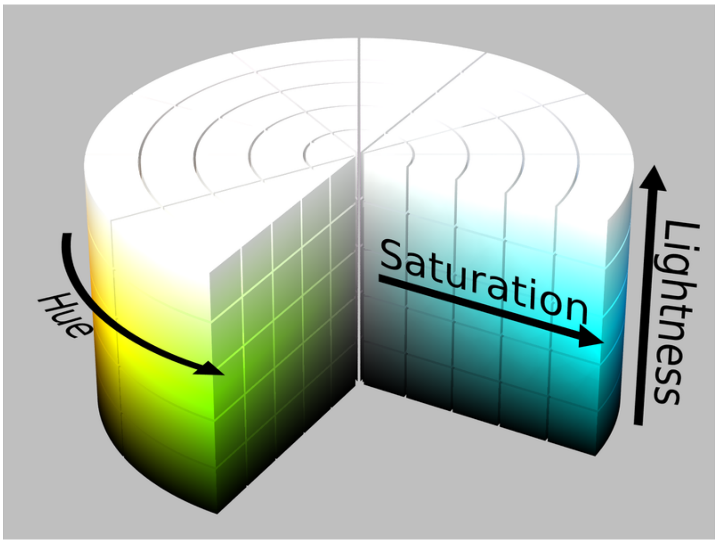

Fig 2 - Cylindre HSL
Source: https://en.wikipedia.org/wiki/File:HSL_color_solid_cylinder_saturation_gray.png

- A quoi correspond la saturation (essayez -100% e +100%)?

La saturation est la purité d'une couleur. Si on ajoute du blanc, du gris ou du noir à une couleur, on la désature.

3. Niveaux de gris, histogrammes et statistiques

3.1. Histogramme

- En considérant les niveaux de gris d'une image comme la réalisation d'une variable aléatoire dont la loi est l'histogramme de l'image, interprétez le résultat.

Soient X et Y deux variables aléatoires; ainsi, l'espérance de la somme de ces deux variables sera égal à la somme des espérances de chaque variable, et la variance de la somme de ces deux variables sera égal à la somme des variances de chaque variable plus deux fois la covariance entre les variables:

$$E(X+Y) = E(X) + E(Y)$$
$$Var(X+Y) = Var(X) + Var(Y) + 2Cov(X,Y)$$

Si les deux variables sont indépendentes, la covariance est nulle:

$$Var(X+Y) = Var(X) + Var(Y)$$
$$std(X+Y) = \sqrt{std^2(X) + std^2(Y)}$$

Si on somme un bruit gaussien a une image, la valeur moyenne autour de lequelle la distribution se concentre ne change pas, mais la variance change, et le plus fort le bruit, le plus il sera dominant. On peut voir ça avec les functions np.mean et np.std, et on peut voir la dominance du bruit ci-dessous.

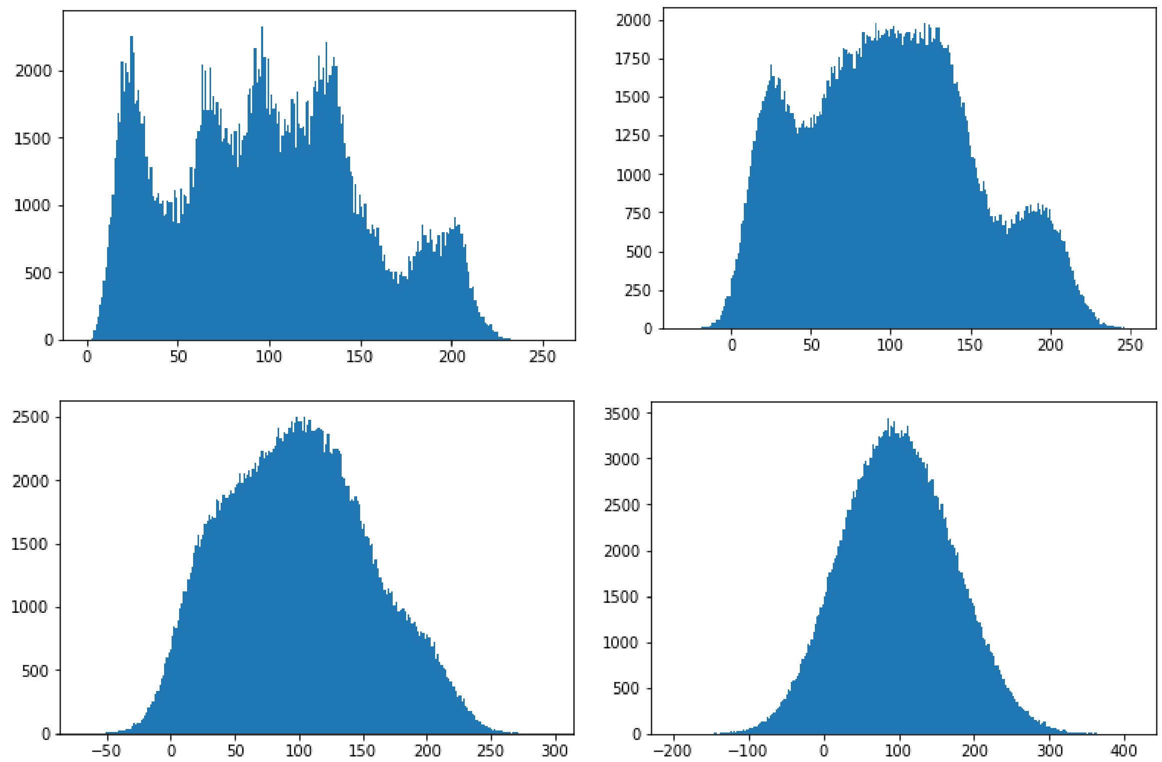

Fig. 3 - Addition d'un bruit gaussien de valeur 0, 10, 20 et 50 a une image.

Quand on somme deux variables aléatoires, le résultat est une convolution des deux variables. Alors, on convolue l'histogramme de l'image avec une fonction gaussienne. Le plus grand l'écart type, plus apparent et plus dominant est le bruit gaussien (si l'écart type est trop petit, c'est comme si on convoluait l'histogramme avec un train d'impulsions).

3.2. Changement de contraste

- L'aspect global de l'image est-il modifié par l'application de fonctions croissantes ? Que se passe-t-il si l'on applique une transformation non-croissante des niveaux de gris?

L'application de fonctions croissants au contraste ne change pas l'aspect global, mais si on applique une transformation non-croissante des niveaux de gris on voit quelque chose comme le négatif de l'image:

Fig. 4 - Transformation de contraste avec une fonction croissante (à gauche) e non-croissante (à droite)

3.3. Égalisation d'histogramme

- Qu'observez-vous sur *imequal*, sur son histogramme et sur son histogramme cumulé?

L'image, aprés l'égalisation d'histogramme, est beaucoup plus bruitée mais plus claire aussi; c'est beaucoup plus facile de percevoir les détails. L'histogramme se rendre éparse et l'histogramme cumulé se rendre presque l'identité.

Fig. 5 - Image avant (à gauche) et après (à droite) l'égalisation d'histogramme

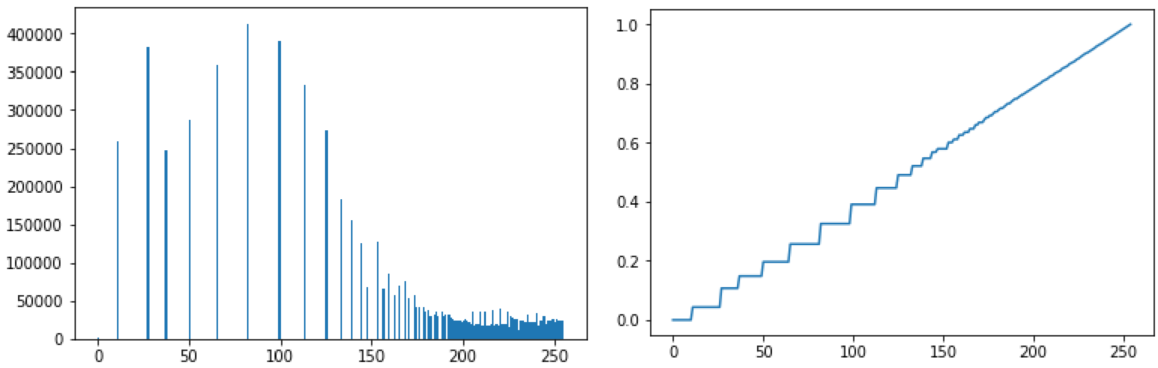

Fig. 6 - L'histogramme (à gauche) et l'histogramme cumulé (à droite) après l'égalisation

3.4. Prescription d'histogramme

- Visualisez la valeur absolue de la différence des images, qu'observe-t-on? Même question après avoir donné à l'une des images l'histogramme de l'autre.

On sait que la différence est plus grand où on voit du blanc, alors on peut dire que sauf pour les objets noirs et pour la source de lumière toute l'image a été bien affecté par la prescription de l'histogramme.


Fig. 7 - Valeur absolue de la différence des images avant de prescrire l'histogramme (à gauche) et après la prescription (à droite)

- A-t-on un moyen plus simple d'obtenir le même résultat (donner le même histogramme aux deux images), dans le cas particulier de ces deux images?

- En vous inspirant des deux lignes de code ci-dessus, donnez un code simples permettant d'égaliser l'histogramme d'une image (le rendre aussi proche que possible d'une fonction constante).

Pour faire cela en utilisant la prescription d'histogramme, on peut créer un vecteur de même taille que l'image et le remplir avec des valeurs uniformes entre 0 et 255, et alors on donne l'histogramme de ce vecteur (un histogramme uniforme) à l'image original. Le code est écrit ci-dessous.

```
#%% une autre forme d'égaliser l'histogramme d'une image
im=skio.imread('images/sombre.jpg')
im=im.mean(axis=2) #on est sur que l'image est grise
viewimage(im)
ind = np.unravel_index(np.argsort(im, axis=None), im.shape)
imequal = np.zeros(im.shape, im.dtype)
vdesire = np.linspace(0,255,imequal.size)
imequal[ind]=np.sort(vdesire, axis=None)
viewimage(imequal)
```

3.5. Dithering

- Appliquez le même seuillage à une version bruitée de l'image originale et visualisez. Que constatez vous?

On peut mieux percevoir les détails et les contours quand on ajoute du bruit avant de seuiller l'image.

- En considérant un pixel de niveau x dans l'image initiale, donnez la probabilité pour que ce pixel soit blanc après ajout de bruit et seuillage. Pourquoi l'image détramée ressemble-t-elle plus à l'image de départ que l'image simplement seuillée?

On note *n* un bruit gaussien centré d'écart type σ. On binarise l'image bruitée avec un paramètre λ:

$$\tilde{I} = \mathbf{1}(I \geq \lambda),\ \text{où } I \text{ est l'image}$$

On suppose que l'image est localement constante de valeur *a*. Si on effecture une moyenne locale, on perçoit:

$$E(\tilde{I}) = E(1(I \geq \lambda)) = P(I \geq \lambda) = P(a + n \geq \lambda)$$

$$E(\tilde{I}) = \int_{\lambda-a}^{+\infty} g_\lambda(x)dx$$

avec g_λ la densité de la variable gaussienne centrée d'écart type σ. Donc:

$$\tilde{I} = F \circ I$$

Étant F une fonction croissante, on a un changement de contraste. On peut interpreter que en faisant la bruitage de l'image avant de la seuillage, on randomise l'erreur de quantification, en évitant les motifs à grande échelle.

3.6. Différences de niveaux de gris voisins

- La distribution des différences vous semble-t-elle obéir a une loi gaussienne ? Pourquoi? Quelle aurait été la forme de l'histogramme si l'on avait considéré la différence entre pixels plus éloignés?
  Oui. Des pixels proches ont en général des valeurs proches, alors la différence entre eux nous donne un niveau de gris clair pour la majorité de l'image. Où on a des bords, on a un grand changement de valeur entre les pixels, mais comme les bords sont une minorité on a peu de valeurs de différence (pixels blancs ou noirs, à cause du overflow). Ça nos donne une loi gaussienne.
  Se on avait considéré la différence entre pixels plus éloignés, on trouverait une courbe plus plate (écart type plus grand), car ainsi on éloigne les bords (et alors on a plus de valeurs noirs et blancs), comme on voir dans la figure ci-dessous.

Fig. 8 - Gradient d'une image vers la direction x, en utilisant des pixels éloignés.

4. Spectre des images et transformation de Fourier

4.1. Visualisation de spectres

- Visualisez les spectres de différentes images en utilisant les options 1 et 2. Que constatez-vous? Que'n déduisez-vous par rapport au spectre d'une image?

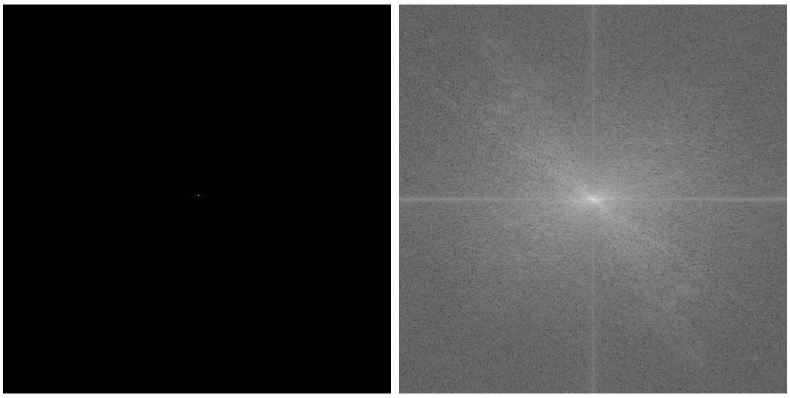

Fig. 9 - Spectre linéaire vs. log.

On voit qu'on ne peut pas extraire beaucoup d'information du spectre linéaire (on ne voit que quelques pixels clairs au milieu de l'image). Dans le spectre em échelle logarithmique, on peut voir plusieurs lignes qui traversent le centre.

- Comment influe l'option hamming sur le spectre de l'image?

Si on active la fênetre de Hamming, on peut voir clairement seulement une ligne diagonale et une droite. Il semble que les lignes qui traversent le centre sont plus floues qu'avant. Le spectre se rend un peu plus foncé aussi.

- Visualisez le spectre de l'image synthétique rayures.tif. Que constatez-vous?
  Peut-on retrouver les caractéristiques des rayures de l'image à partir de son spectre?
  Expliquez la différence entre la visualisation avec e sans l'option hamming.

On voit des multiples lignes droites diagonales parallèles et verticales parallèles. On voit aussi une seule ligne horizontale qui traverse le centre de l'image.

On peut retrouver les caractéristiques originales en utilisant la Transformée de Fourier Discrete Inverse (TFD inverse).

Avec l'option hamming, le spectre se rend beaucoup plus foncé. Les lignes diagonales se rendent plus visibles, pendant que les lignes verticales sont plus faibles (mais plus épaisses aussi).

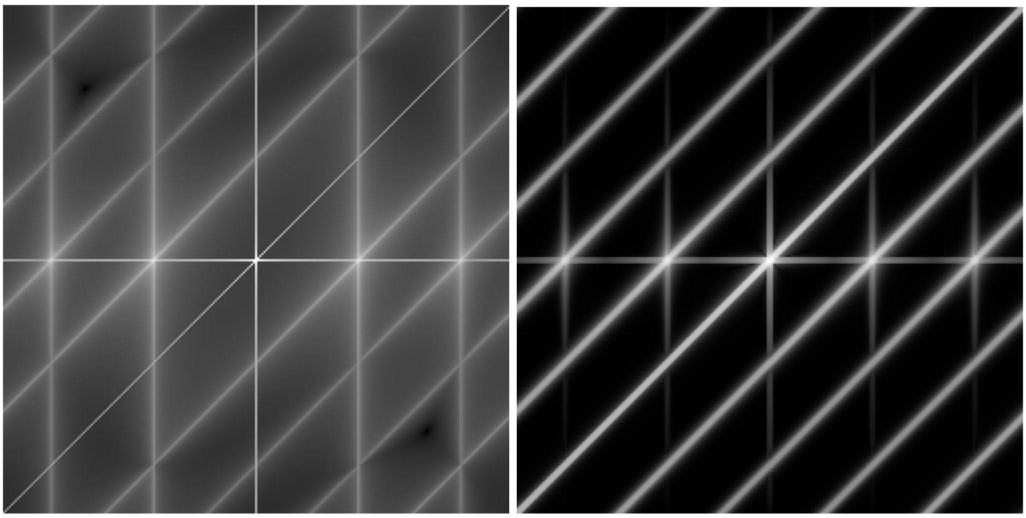

Fig. 10 - Spectre sans(à gauche) et avec (à droite) la fenêtre de Hamming

- Quel effet a le sous-échantillonnage sur le spectre?

Le spectre de l'image sous-échantillonnée semble beaucoup plus bruité, avec une granularité grossière des pixels.

4.2. Ringing

- Visualisez l'image résultante, ainsi que son spectre. Que constatez-vous?

On voit qu'il y a du ringing autour de tous les bords, comme on peut voir dans l'image ci-dessous. Dans le spectre, on voit que les hautes fréquences - plus distantes du centre - sont toutes supprimées (on voit un bord noir gros autour du spectre).

Fig. 11 - Zoom du chapeau de Lena après application d'un filtre passe-bas parfait

- Mêmes questions en utilisant la commande filtergauss.

Après l'application du filtre gaussien, on ne voit plus de *ringing*, mais l'image a plus de flou. Dans le spectre, les hautes fréquences ne sont pas supprimées, mais elles sont atténuées (le plus loin du centre, le plus foncé est le spectre).

- Visualisez les deux masques [...]. Quelle différence constatez-vous, en particulier quelle conséquence a la discontinuité de la transformée de Fourier sur la vitesse de décroissance du filtre spatial correspondant?

La discontinuité cause le *ringing* autour des bords. C'est la même chose qu'on peut voir quand on faire la transformée de Fourier d'un signal carré, on a des oscillations autour du point de discontinuité. C'est ce qu'on appelle l'effet Gibbs.

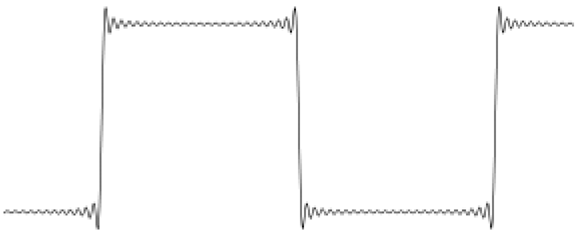

Fig. 12 - Effet Gibbs sur