

Travaux pratiques 1 - IMA201- Réponses

Júlia Togashi de Mirada

2. Visualisation et utilisation de gimp

2.1. Zooms

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

Il effectue le zoom grâce à une méthode d'interpolation. (Calculer la valeur de l'image en un point qui ne fait pas partie de la grille d'origine).

Quelle hypothèse pouvez-vous faire sur la génération de lena_petit.tif ?

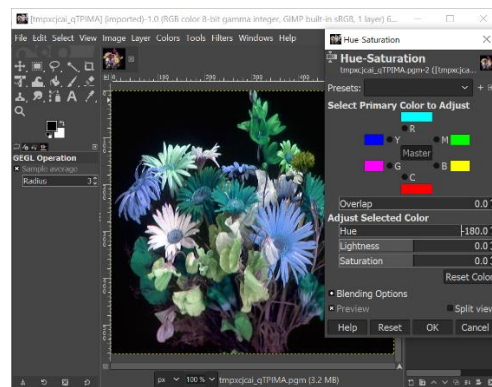
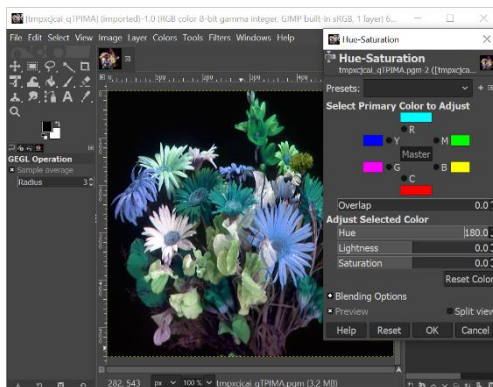
```
im=skio.imread("images/lena.tif")  
viewimage(im,normalise=False)
```

Comme on peut le voir, l'image Lena_petit est beaucoup plus nette que Lena après un zoom. L'image réduite de Lena a probablement été obtenue en faisant la moyenne d'un carré de $n \times n$ pixels (dans ce cas, car c'est la moitié, 2×2). Afin d'obtenir Lena_petit, une pondération dans cette moyenne a probablement été utilisée, afin d'obtenir un meilleur résultat.



Lena avec un réduction de taille d'un facteur deux (Aucune méthode d'interpolation) VS Lena_petit

2.2. Espace couleurs



Transformation des fleurs jaunes en fleurs bleues avec le bouton Hue

Comprenez-vous pourquoi les deux positions extrêmes de ces boutons font la même transformation ?

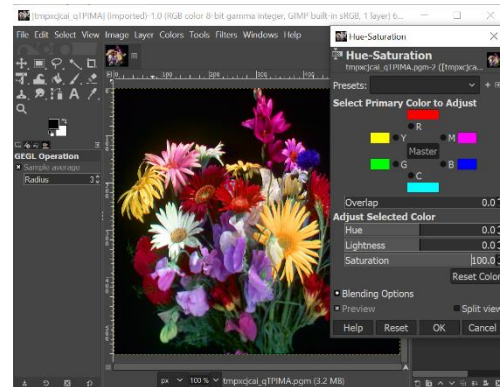
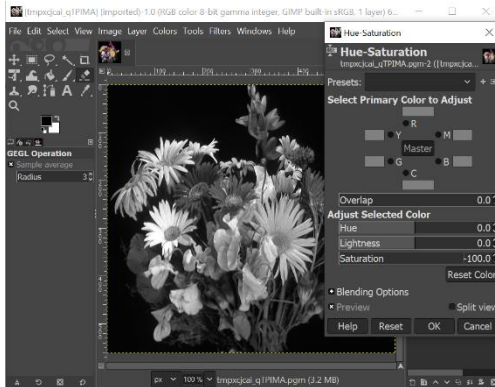


Parce que le « Hue » c'est comme un cercle.
Donc $-180 \rightarrow 0^\circ$ et $180 \rightarrow 360^\circ$.
De cette manière, la couleur aux extrémités sera la même.

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

La saturation c'est l'attribut permettant d'estimer la proportion de couleur chromatiquement pure contenue dans la sensation totale. Caractérise le côté plus ou moins "délavé", degré de mélange de la longueur d'onde prédominante avec le blanc.

Donc à -100, problème avec peu de couleur (noir et blanc) et à 100 avec des couleurs vibrantes



Saturation -100% et +100%, respectivement

3. Niveaux de gris, histogrammes et statistiques

3.1. Histogramme

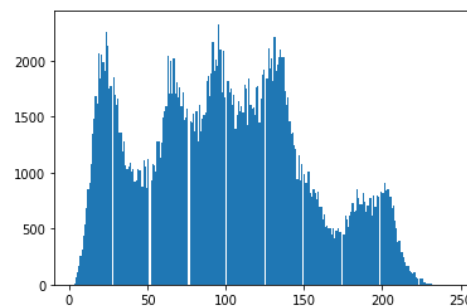


Image Lena et son histogramme

```
imcol=skio.imread('images/after.jpg')  
imcol=np.float32(imcol)  
rouge=imcol[:, :, 0]  
plt.hist(rouge.reshape((-1,)),bins=255)
```

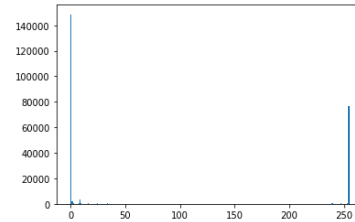
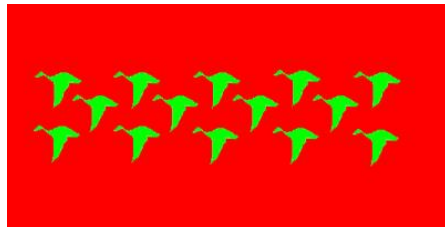


Image after et son histogramme

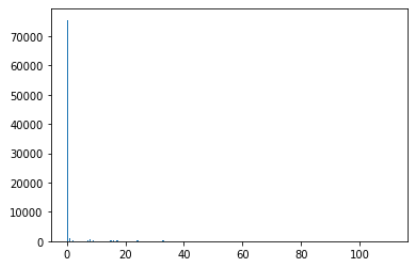
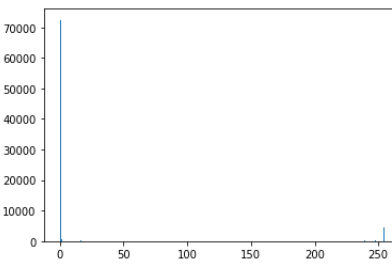
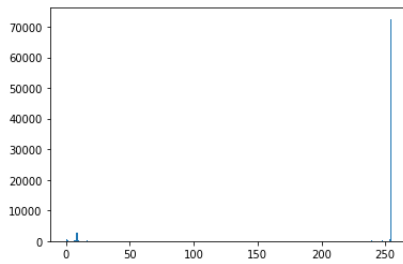
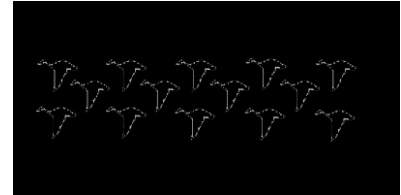
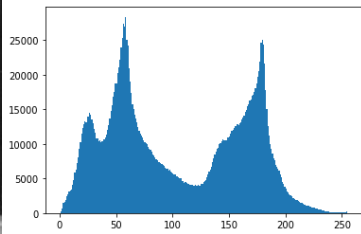
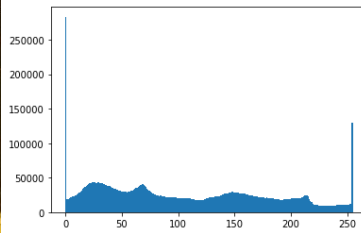


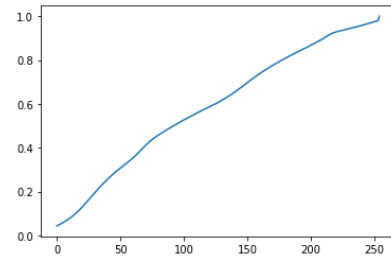
Image after et respectif histogramme en bas, quand extrayons le canal rouge ; vert ; bleu



Quand on regarde le résultat de l'histogramme quand extrayons le canal rouge ; vert ; bleu, on observe que le canal rouge a beaucoup des valeurs à 250 et le bleu à 0. Donc quand on fait la moyenne, il est logique que le résultat obtenu.

Image chat avant et après faire la moyennant les canaux couleur

```
im=np.float32(im)
(histo,bins)=np.histogram(im.reshape((-1,)),np.arange(0,256))
histo=histo/histo.sum()
histocum=histo.cumsum()
plt.plot(histocum)
```



L'histogramme cumulé d'image Chat

```
im=skio.imread('images/spirale.jpg')
imbr=noise(im,10)
viewimage_color(imbr,normalise=False)
plt.hist(im.reshape((-1,)),255)
plt.show()
plt.hist(imbr.reshape((-1,)),255)
plt.show()
```

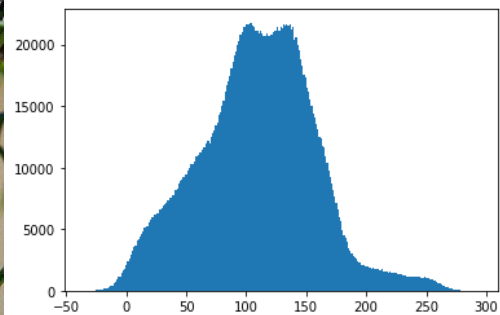
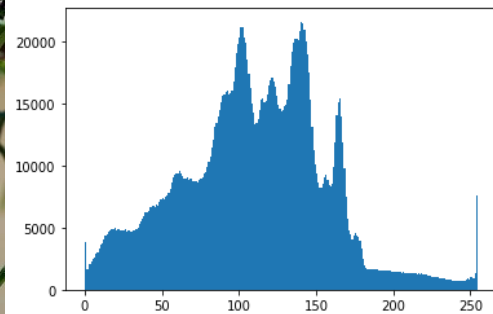


Image et histogramme de spirale avant et après ajouter le bruit

Comme on peut le voir dans l'image ci-dessus, quand nous ajoutons un bruit à la image (la réalisation d'une variable aléatoire dont la loi est l'histogramme de l'image), et cette bruit a un forme Gaussian, le histogramme convolué avec une gaussienne centrée d'écart-type σ . De cette façon, l'histogramme semble adouci.

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 ?

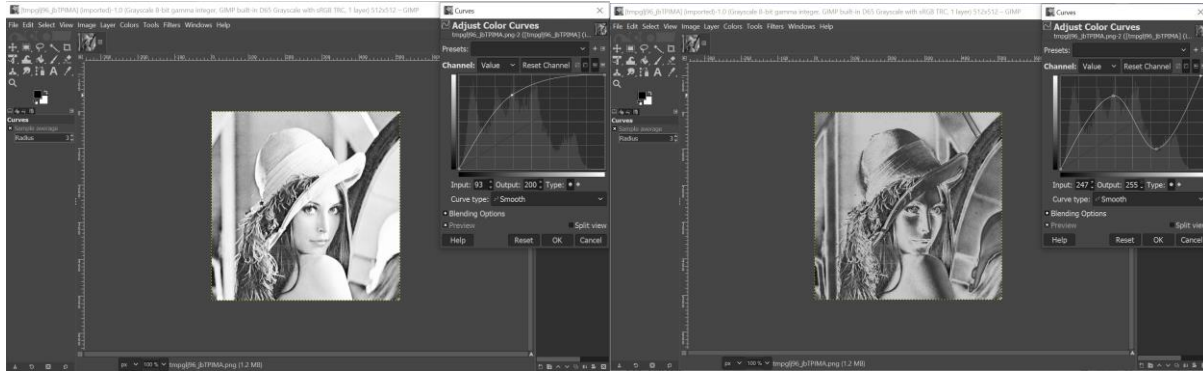
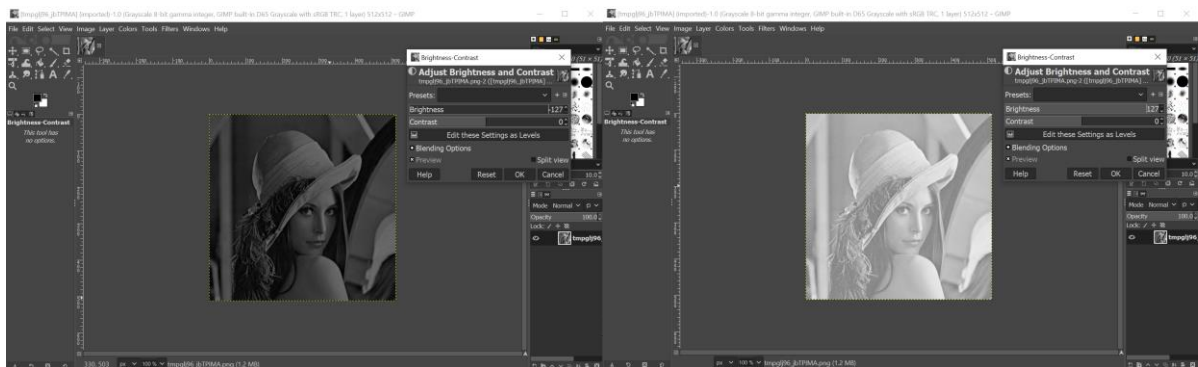
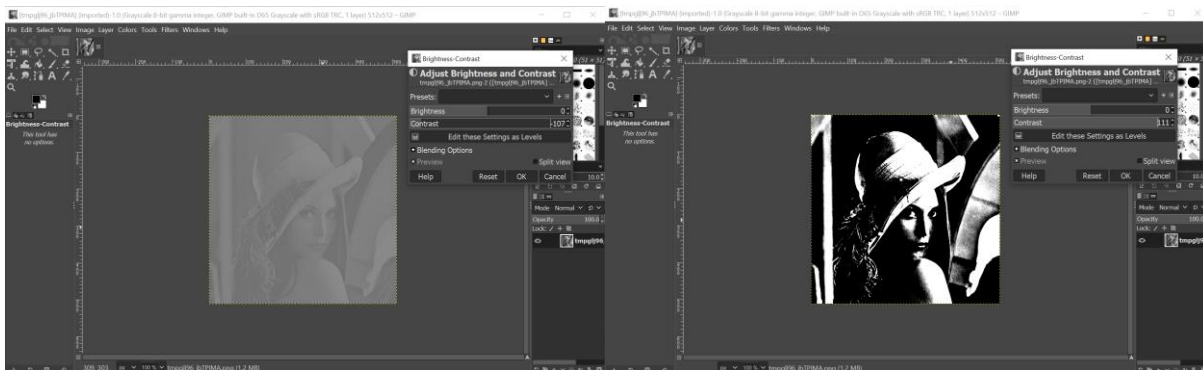


Image Lena quand on applique une fonction croissante et quand on applique une transformation non-croissante

Oui, l'aspect global est plus clair, mais la perception du contenu d'une image varie peu si l'on applique une fonction croissante à l'image. Au cas où la transformation est non-croissante la perception du contenu d'une image varie beaucoup, le résultat dans ce cas ressemble au négatif de l'image.



Résultat quand on met la luminosité au minimum et au maximum.



Résultat quand on met le contraste au minimum et au maximum.

3.3. Egalisation d'histogramme

```
im=skio.imread('images/sombre.jpg')
im=im.mean(axis=2)
viewimage(im)
plt.hist(im.reshape((-1,)),bins=255)
(histo,bins)=np.histogram(im.reshape((-1,)),np.arange(0,256))
histo=histo/histo.sum()
histocum=histo.cumsum()
plt.plot(histocum)
imequal=histocum[np.uint8(im)]
viewimage(imequal)
plt.hist(imequal.reshape((-1,)),bins=255)
(histo,bins)=np.histogram(imequal.reshape((-1,)) *256,np.arange(0,256))
histo=histo/histo.sum()
histocum=histo.cumsum()
plt.plot(histocum)
```



Image « sombre », histogramme et histogramme cumulé avant l'égalisation d'histogramme

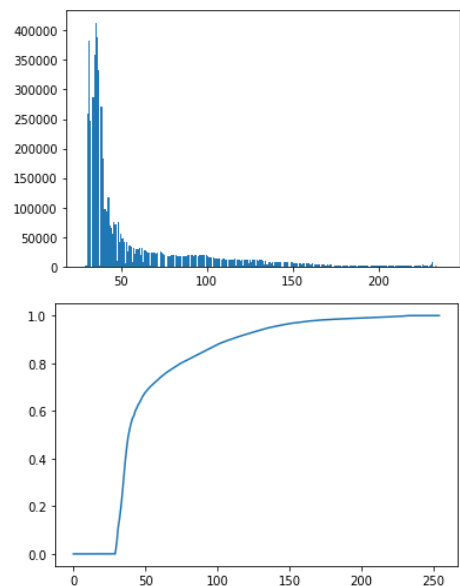
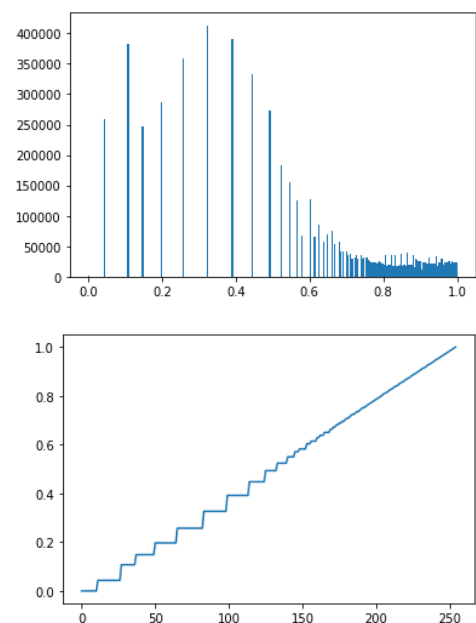


Image « sombre », histogramme et histogramme cumulé après l'égalisation d'histogramme



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

L'égalisation d'histogramme est une action pour changer la distribution des valeurs d'occurrence dans un histogramme, permettant une réduction des différences accentuées et donc, en particulier dans les images, accentuant des détails non visibles auparavant.

Donc l'histogramme et histogramme cumulé sont plus douce et la photo c'est plus clair, avec plus de détail.

3.4. Prescription d'histogramme

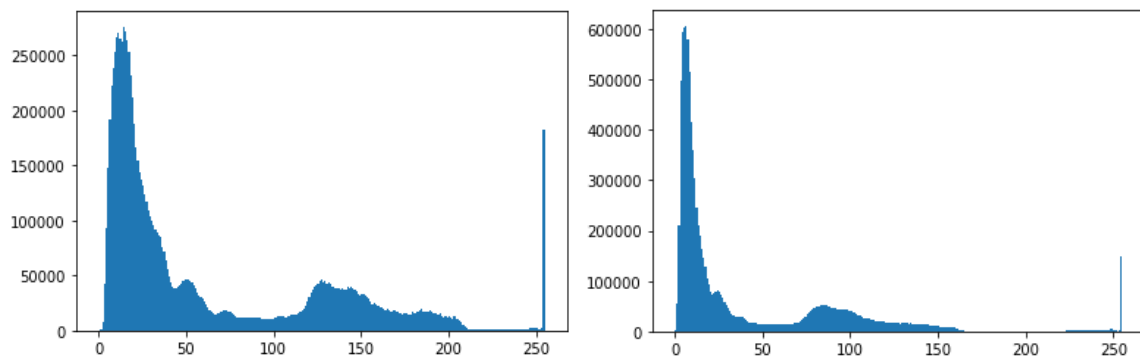
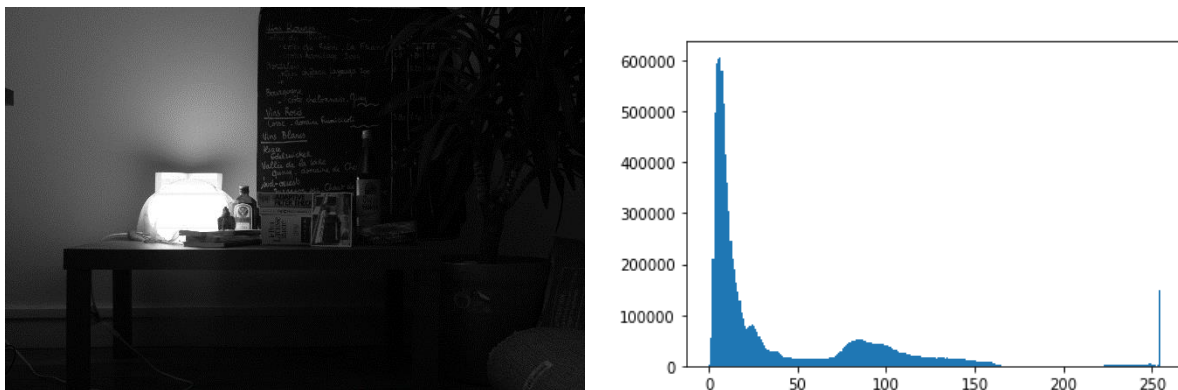


Image et histogramme de vue1.tif et vue2.tif respectivement



Résultat après la Prescription d'histogramme



Différence des images, qu'on observe avant et après avoir donné à l'une des images l'histogramme de l'autre

Nous pouvons voir que la différence avant c'est plus grand, parce qu'une image c'est plus clair que l'autre. Après la différence c'est très petit.

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 ?

Un moyen très simple de donner à u le même histogramme que v :

```
[x,index] = sort(u( :)) ;  
u(index) = sort(v( :)) ;
```

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

```
u=skio.imread(r'C:\Users\jutogashi\Documents\Telecom_2A\IMA\IMA201\images\vue1.tif')  
ind=np.unravel_index(np.argsort(u, axis=None), u.shape)  
unew=np.zeros(u.shape,u.dtype)  
unew[ind]=np.arange(0,255,255/(u.shape[0]*u.shape[1]))  
viewimage(unew)
```

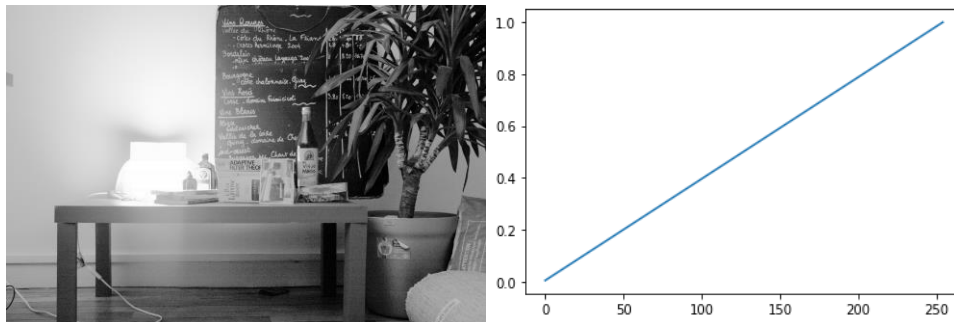


Figure vue1 avec l'histogramme régularise

3.5. Dithering



Image Lena avec l'effet d'une quantification égal à 2 ;4 et 8, respectivement

Nous pouvons le constater, comme prévu, en augmentant de la quantification, améliore l'image.



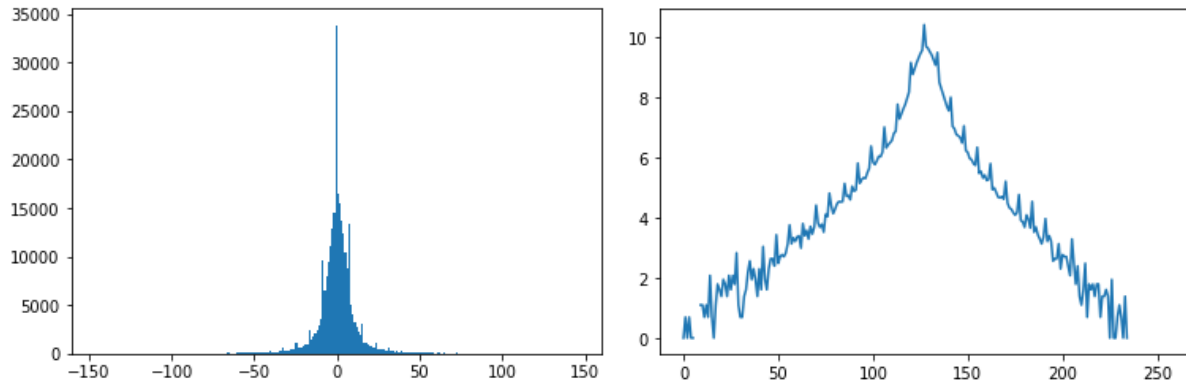
Utilisation du seuil dans l'image Lena et dans l'image Lena avec bruit.

L'image avec bruit évidemment se ressemble plus à l'image original que sans le bruit.

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 ?

Cela semble plus parce que lors de l'ajout de bruit, nous ajoutons des pixels proches qui auront des couleurs différentes (plus claires ou plus foncées), alors qu'avant nous avions une image constante par parties. Ainsi, avant qu'une grande zone soit complètement noire ou blanche, maintenant les pixels qui deviennent noirs ou blancs sont rapprochés, créant une illusion de nuances de gris.

3.6. Différences de niveaux de gris voisins



Histogramme e logaritme du histogramme la différence de niveau de gris entre deux pixels adjacents de l'image Lena.

La distribution des différences vous semble-t-elle obéir à une loi gaussienne ? Pourquoi ? Quelle aurait été la forme de l'histogramme si l'on avait considéré la différence entre pixels plus éloignés ?

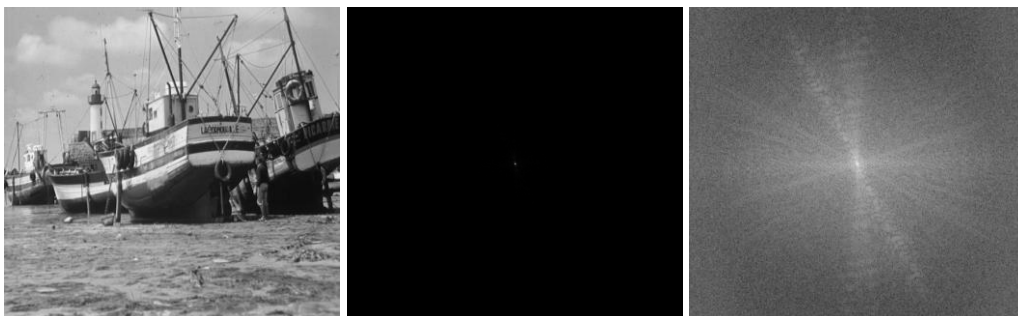
Oui. Parce que la plus grande différence entre les petits voisins réside dans le contour de l'image. Par conséquent, la plupart d'entre eux ont une différence de moyenne de 0, l'écart type correspondant au contour. Je pense que c'est aussi une distribution gaussienne, mais avec un l'écart type plus grand.

4 Spectre des images et transformation de Fourier

4.1 Visualisation de spectres



Le spectre de l'image chapeau avec l'option =1 on affiche l'intensité de manière linéaire et l'option =2 on affiche le log, respectivement.

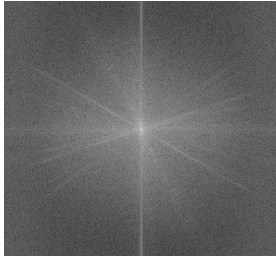


Le spectre de l'image bat avec l'option =1 on affiche l'intensité de manière linéaire et l'option =2 on affiche le log, respectivement.

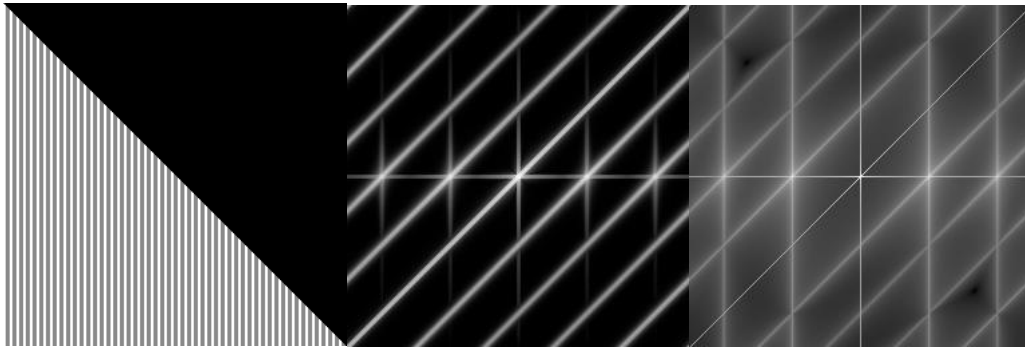
Que constatez-vous ? Qu'en déduisez-vous par rapport au spectre d'une image ?

On voit que le spectre de l'image du bateau, par exemple, semble plus net que celui du chapeau, avec des rayons bien définis.

Nous avons que les coefficients situés du centre sont des fréquences basses (zones dégradées) et les en périphérie du spectre correspondent aux hautes fréquences (contours, détails). Donc on peut dire que l'image du bateau a plus de détails que du chapeau.



Ici on a le spectre de l'image du bateau sans l'option hamming. Nous pouvons voir qu'ici les rayons sont plus clairs qu'avec l'option.



L'image artificiel rayon. Son spectre avec l'option hamming et sans.

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 et sans l'option hamming ? Quel effet a le sous-échantillonnage sur le spectre ?

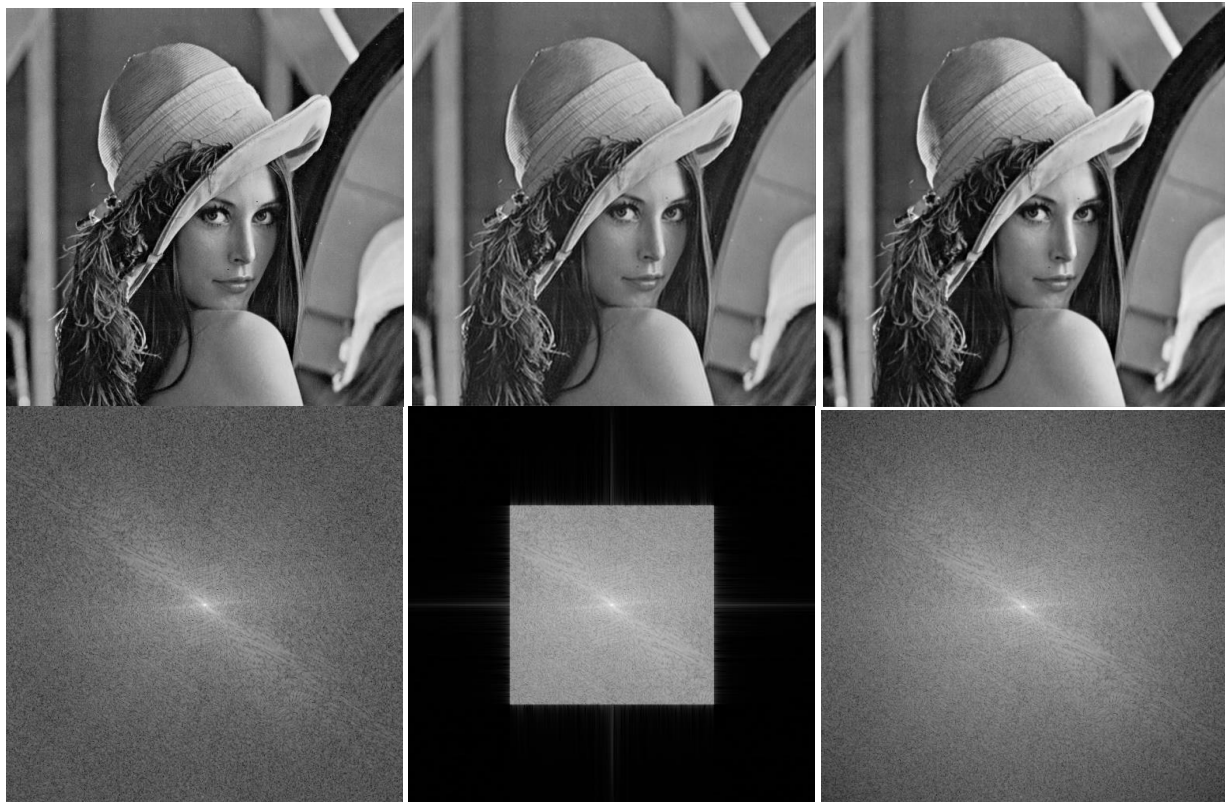
Le spectre de l'image a des rayons très forts aussi.

Oui, on peut. Quand une image a de fréquence, cette image ne contenant qu'une seule fréquence, son spectre ne contient que 2 valeurs non nulles. Comme demi de l'image c'est noir, nous obtenue la fréquence des rayons et le contour avec le noir.

L'image sans hamming se trouve moins clairs, avec moins de contraste entre les fréquences.

Quand on trouve une image avec le sous-échantillonnage nous pouvons percevoir que l'image c'est plus clair, qui représente une augmentation de la fréquence globale de l'image.

4.2 Ringing



L'image Lena originale, avec de filtre passe bas, avec le filtre de gauss et ses respectif spectre

Comme prévu, quand on utilise le filtre passe bas, on retire des fréquences hautes dans l'image, donc ils disparaissent du spectre. Dans le filtre de Gauss, on peut percevoir que le spectre c'est plus constant, qui est logique comme un filtre capable de réduire le niveau de bruit d'un signal d'entrée, afin de diminuer la distorsion d'une image, il diminue la variation de fréquence.

Visualisez les deux masques (sous le répertoire images) masque_bas_centre.tif (pour le filtrage passe-bas parfait) et masque_gauss_centre.tif (pour le filtrage gaussien). 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 ?

Le masque_bas_centre montre que on retire les fréquences plus hautes et le masque_gauss_centre montre que on retire comme une fonction gaussienne. LA conséquence de la discontinuité c'est qu'on voit apparaître des « effets de rebond » (la duplication des contours), comme proche au chapeau de la femme.