

(2) Filtrage dans le plan (complexe) de Fourier

- On utilise le théorème de convolution de la TF :

$$f * h \xrightarrow{TF} \hat{f} \cdot \hat{h}$$

$$f \cdot h \xrightarrow{TF} \hat{f} * \hat{h}$$

avec f = image, h = filtre $\Rightarrow \hat{h}$ = fonction de transfert de h

h = filtre (et \hat{h} sert de masque dans le plan de Fourier)

L'idée ici est de modeler \hat{h} (à travers son module) afin de modifier la distribution des fréquences spatiales présentes dans l'image f .

- Filtrage passe-bas

On fait diminuer les valeurs de \hat{h} au fur et à mesure que l'on s'éloigne du centre $(0,0)$ des fréquences spatiales, favorisant ainsi les plus basses fréquences en atténuant les plus hautes.

Cette distribution peut-être brutale (on « coupe » alors les fréquences plus hautes qu'une certaine

fréquence de coupure à l'aide d'une porte de largeur $(2 u_c, 2 v_c)$ autour du frequel $(0,0)$, où : u_c est la fréquence de coupure en u , v_c celle en v .
Applications : supprimer les détails fins, réduire le bruit.

Remarque : on prend en général des fonctions de transfert \hat{h} autour du frequel $(0,0)$.

Exemple : revisite du filtrage Gaussien (où l'on cherche à faire le filtrage Gaussien du chapitre 3 mais en passant par Fourier...)

```
>> dim=128;
>> l=zeros(dim,dim);
>> l(:,1:dim/2) = 0.6;
>> figure, colormap(gray)
>> subplot(3,2,1), imagesc(l), colorbar, title('img')
—
>> lchap = fft2(l);
>> whos lchap;
>> subplot(3,2,2), imagesc(abs(fftshift(lchap)).^0.5),
colorbar, title('|FFT(image)|')
—
>> h = fspecial('gaussian', dim, 0.849);
>> subplot(3,2,3), imagesc(h.^0.5), colorbar,
title('Gauss')
—
>> hchap = fft2(h);
>> whos hchap;
```

```
>> subplot(3,2,4), imagesc(abs(fftshift(hchap)).^0.5),  
colorbar, title('|FFT(Gauss)|')
```

```
>> lchapfilt = hchap.*lchap;
```

```
>> whos lchapfilt
```

```
>> subplot(3,2,6),
```

```
imagesc(abs(fftshift(lchapfilt)).^0.5), colorbar, title('|  
FFT(image filtrée)|')
```

```
>> lfilt = ifftshift(ifft2(lchapfilt));
```

```
>> whos(lfilt)
```

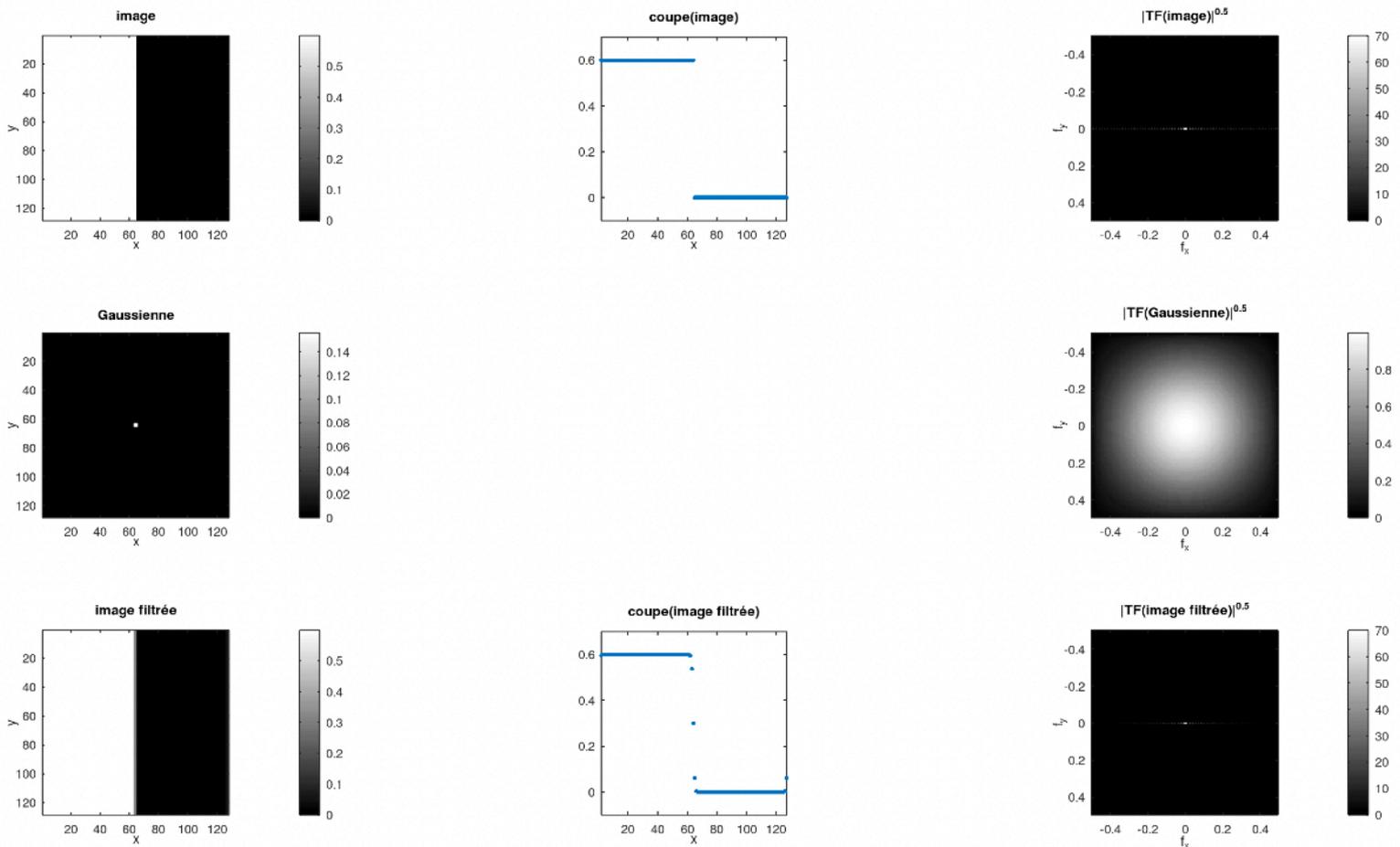
```
>> subplot(3,2,5), imagesc(real(lfilt)), colorbar,  
title('image filtrée')
```

```
1 clear  
2 close all  
3 pkg load image  
4  
5 % image  
6 dim=128; I=zeros(dim,dim); I(:,1:dim/2)=0.6;  
7 fx=((0:dim-1)-dim/2)/dim; fy=fx;  
8  
9 figure, colormap('gray')  
10  
11 subplot(3,3,1), imagesc(I)  
12 colorbar, axis('square')  
13 title('image'), xlabel('x'), ylabel('y')  
14  
15 subplot(3,3,2), plot(I(64,:),'.')  
16 axis('square')  
17 title('coupe(image)'), xlim([2,dim-1]), ylim([- .1 .7]), xlabel('x')  
18  
19 % FFT(image)  
20 Ichap=fft2(I); Ichapmod=abs(fftshift(Ichap));  
21 subplot(3,3,3), imagesc(fx,fy,Ichapmod.^5)  
22 colorbar, axis('square')  
23 title('|ITF(image)|^{0.5}'), xlabel('f_x'), ylabel('f_y')  
24  
25 % filtre  
26 h=fspecial('gaussian',dim,.849);  
27 subplot(3,3,4), imagesc(h)  
28 colorbar, axis('square')  
29 title('Gaussienne'), xlabel('x'), ylabel('y')  
30  
31 % => fct de transfert  
32 hchap=fft2(h); hchapmod=abs(fftshift(hchap));  
33 subplot(3,3,6), imagesc(fx,fy,hchapmod.^5)  
34 colorbar, axis('square')  
35 title('|ITF(Gaussienne)|^{0.5}'), xlabel('f_x'), ylabel('f_y')  
36
```

```

31 % => fct de transfert
32 hchap=fft2(h); hchapmod=abs(fftshift(hchap));
33 subplot(3,3,6), imagesc(fx,fy,hchapmod.^.5)
34 colorbar, axis('square')
35 title('|TF(Gaussienne)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
36
37 % filtrage dans l'espace de Fourier
38 Ichapfilt=hchap.*Ichap; Ichapfiltmod=abs(fftshift(Ichapfilt));
39 subplot(3,3,9), imagesc(fx,fy,Ichapfiltmod.^.5)
40 colorbar, axis('square')
41 title('|TF(image filtrée)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
42
43 % retour vers l'espace réel
44 Ifilt=ifftshift(ifft2(Ichapfilt));
45 subplot(3,3,7), imagesc(real(Ifilt))
46 axis('square'), colorbar
47 title('image filtrée'), xlabel('x'), ylabel('y')
48
49 subplot(3,3,8), plot(real(Ifilt(64,:)),'.')
50 axis('square'), xlim([2,dim-1]), ylim([-1 .7])
51 title('coupe(image filtrée)'), xlabel('x')

```

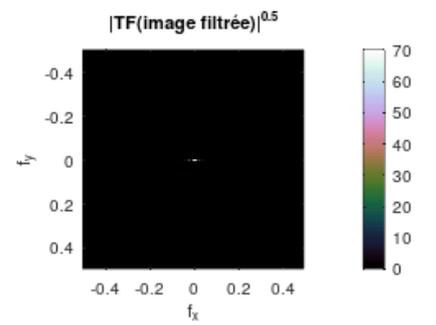
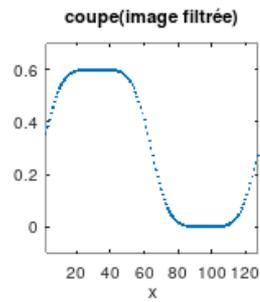
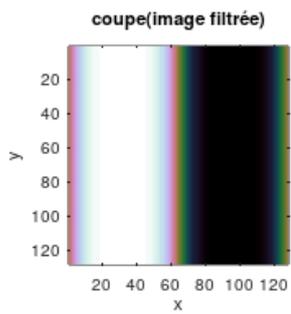
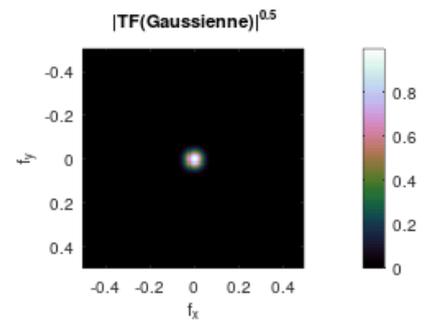
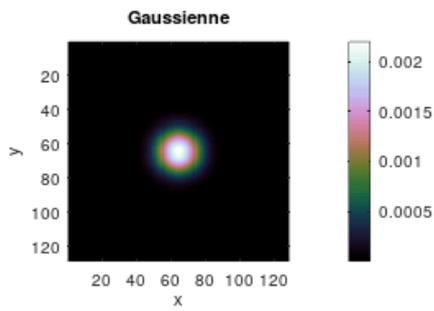
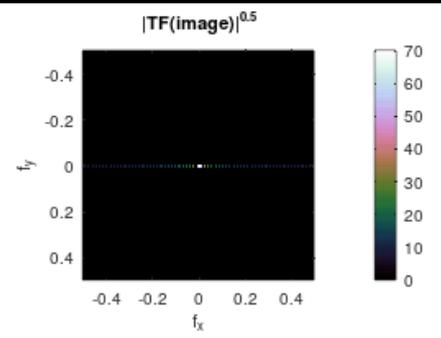
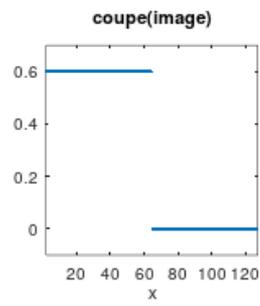
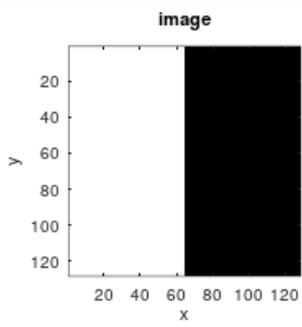


Exercice 5bis : Reprendre l'exemple ci-avant et diminuer la largeur de \hat{h} d'un facteur 10. Que se passe-t-il ?

```

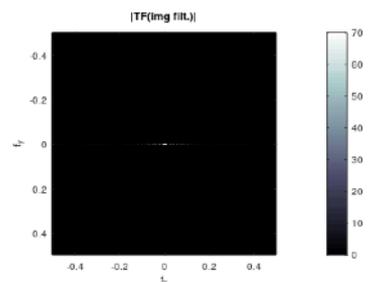
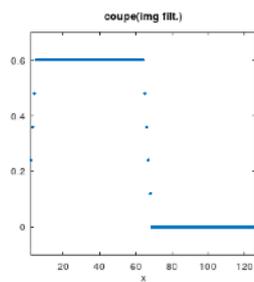
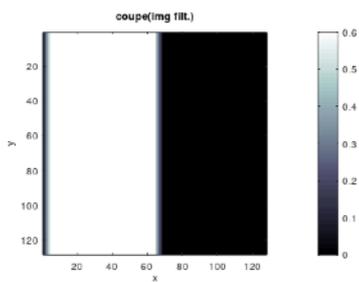
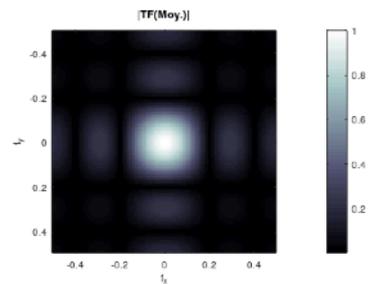
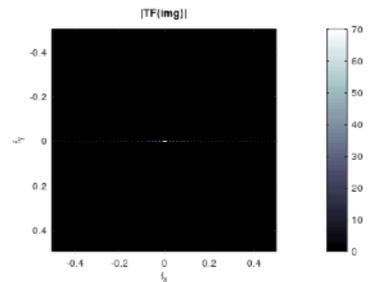
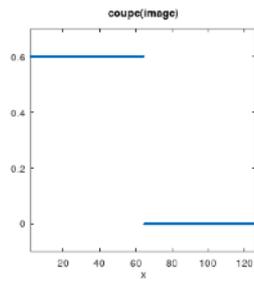
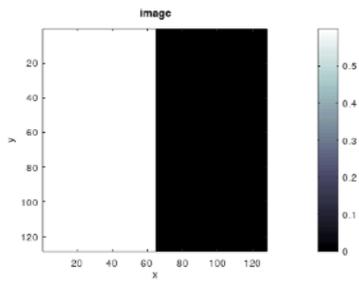
1 clear
2 close all
3 pkg load image
4
5 % image
6 dim=128; I=zeros(dim,dim); I(:,1:dim/2)=0.6;
7 fx=((0:dim-1)-dim/2)/dim; fy=fx;
8
9 figure, colormap('cubehelix')
10
11 subplot(3,3,1), imagesc(I)
12 colorbar, axis('square')
13 title('image'), xlabel('x'), ylabel('y')
14
15 subplot(3,3,2), plot(I(64,:),'.')
16 axis('square')
17 title('coupe(image)'), xlim([2,dim-1]), ylim([-0.1 .7]), xlabel('x')
18
19 % FFT(image)
20 Ichap=fft2(I); Ichapmod=abs(fftshift(Ichap));
21 subplot(3,3,3), imagesc(fx,fy,Ichapmod.^5)
22 colorbar, axis('square')
23 title('ITF(image)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
24
25 % filtre
26 h=fspecial('gaussian',dim,.849*10);
27 subplot(3,3,4), imagesc(h)
28 colorbar, axis('square')
29 title('Gaussienne'), xlabel('x'), ylabel('y')
30
31 % => fct de transfert
32 hchap=fft2(h); hchapmod=abs(fftshift(hchap));
33 subplot(3,3,6), imagesc(fx,fy,hchapmod.^5)
34 colorbar, axis('square')
35 title('ITF(Gaussienne)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
36
37 % filtrage dans l'espace de Fourier
38 Ichapfilt=hchap.*Ichap; Ichapfiltmod=abs(fftshift(Ichapfilt));
39 subplot(3,3,9), imagesc(fx,fy,Ichapfiltmod.^5)
40 colorbar, axis('square')
41 title('ITF(image filtrée)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
42
43 % retour vers l'espace réel
44 Ifilt=ifftshift(ifft2(Ichapfilt));
45 subplot(3,3,7), imagesc(real(Ifilt))
46 axis('square'), colorbar
47 title('coupe(image filtrée)'), xlabel('x'), ylabel('y')
48
49 subplot(3,3,8), plot(real(Ifilt(64:)),'.')
50 axis('square'), xlim([2,dim-1]), ylim([-0.1 .7])
51 title('coupe(image filtrée)'), xlabel('x')

```



Exercice 6 : Faire de même avec la moyenne glissante, en 3x3 puis 5x5.

```
1 clear
2 close all
3 pkg load image
4
5 % image
6 dim=128; I=zeros(dim,dim); I(:,1:dim/2)=0.6; % image = marche 0.6/0
7 fx=((0:dim-1)-dim/2)/dim; fy=fx; % étalonnage des fréquences
8
9 figure, colormap('bone') % colormap "bone" (vor help)
10
11 subplot(3,3,1), imagesc(I)
12 colorbar, axis('square')
13 title('image'), xlabel('x'), ylabel('y')
14
15 subplot(3,3,2), plot(I(dim/2,:),'.')
16 title('coupe(image)'), xlabel('x')
17 axis('square'), xlim([2,dim-2]), ylim([-1 .7])
18
19 % FFT(image)
20 Ichap=fft2(I); % FFT(image)
21 Ichapmod=abs(fftshift(Ichap)); % module réordonné
22 subplot(3,3,3), imagesc(fx,fy,Ichapmod.^5)
23 colorbar, axis('square')
24 title('|ITF(img)|'), xlabel('f_x'), ylabel('f_y')
25
26 % filtre
27 nn=5; h=fspecial('average',nn); % moyenne glissante 3x3 ou 5x5
28 % on peut utiliser aussi ones(nn,nn).
29 % et on peut ensuite recentrer le filtre h dans un tableau dim*dim "à la main,
30 % ou utiliser la commande "padarray", mais ce n'est pas nécessaire puisque l'on
31 % pourra tout aussi bien faire ça au vol avec la commande FFT2.
32
33 % => fct de transfert
34 hchap=fft2(h,dim,dim); % masque=FFT(filtre)
35 hchapmod=abs(fftshift(hchap)); % module
36 subplot(3,3,6), imagesc(fx,fy,hchapmod)
37 colorbar, axis('square')
38 title('|ITF(Moy.)|'), xlabel('f_x'), ylabel('f_y')
39
40 % filtrage dans l'espace de Fourier
41 Ichapfilt=Ichap.*hchap; % filtrage
42 Ichapfiltmod=abs(fftshift(Ichapfilt)); % module du résultat dans Fourier
43 subplot(3,3,9), imagesc(fx,fy,Ichapfiltmod.^5)
44 colorbar, axis('square')
45 title('|ITF(img filt.)|'), xlabel('f_x'), ylabel('f_y')
46
47 % retour vers l'espace réel
48 Ifilt=real(ifft2(Ichapfilt)); % image filtrée
49 subplot(3,3,7), imagesc(Ifilt)
50 colorbar, axis('square')
51 title('coupe(img filt.)'), xlabel('x'), ylabel('y')
52
53 subplot(3,3,8), plot(Ifilt(64,:),'.')
54 title('coupe(img filt.)'), xlabel('x')
55 axis('square'), xlim([2,dim-2]), ylim([-1 .7])
```

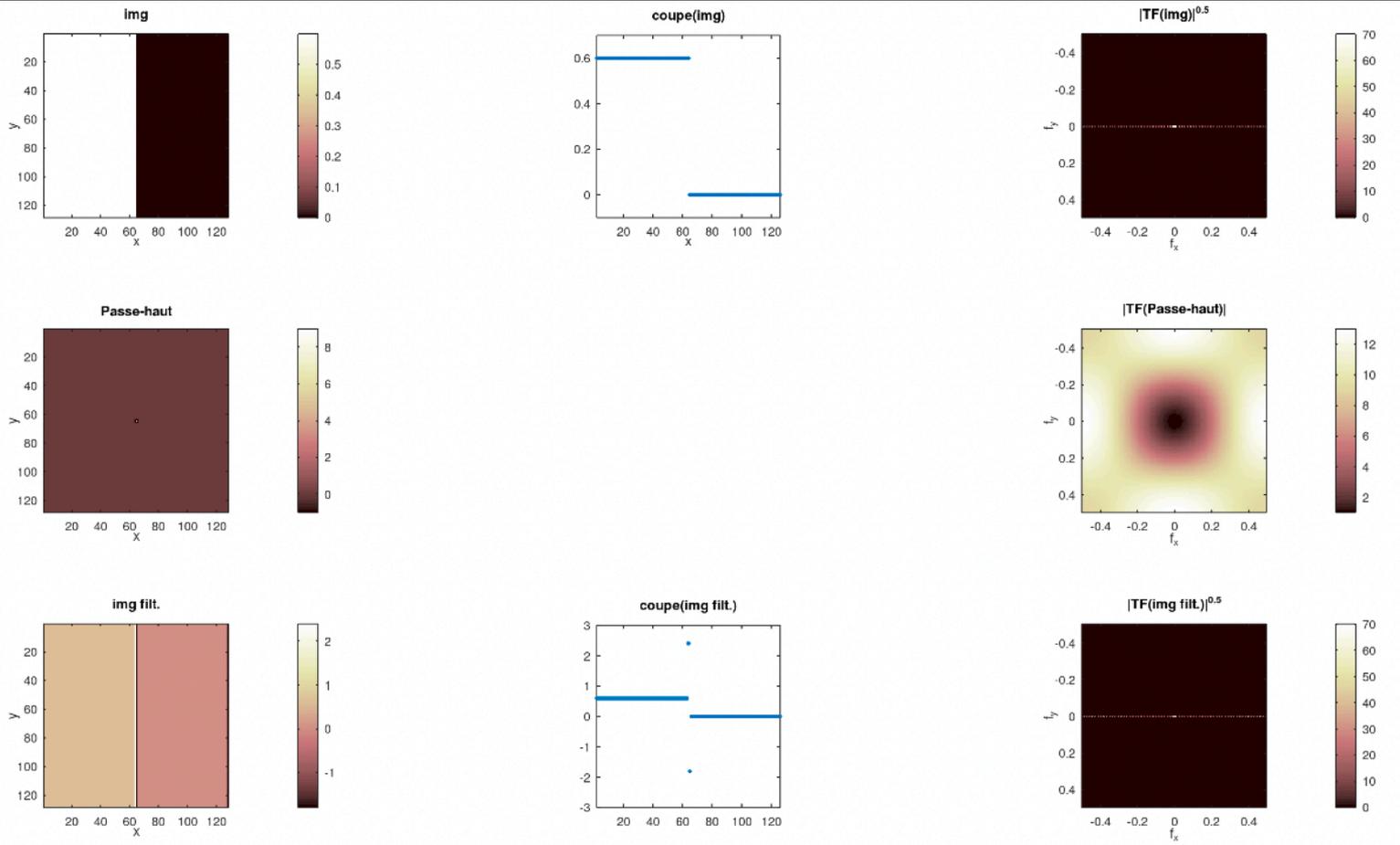


Exercice 7 : Faire de même avec le filtre passe-haut vu au chapitre 3. -1 -1 -1

-1 9 -1

-1 -1 -1

```
1 clear
2 close all
3 pkg load image
4
5 % image
6 dim=128; I=zeros(dim,dim); I(:,1:dim/2)=0.6; % image=marche 0.6/0
7 fx=((0:dim-1)-dim/2)/dim; fy=fx; % étalonnage des fréquences
8
9 figure, colormap('pink')
10
11 subplot(3,3,1), imagesc(I), colorbar, axis('square')
12 title('img'), xlabel('x'), ylabel('y')
13
14 subplot(3,3,2), plot(I(64,:),'.'), axis('square')
15 title('coupe(img)'), xlim([2,dim-2]), ylim([-1 .7]), xlabel('x')
16
17 % FFT(image)
18 Ichap=fft2(I);
19 Ichapmod=abs(fftshift(Ichap));
20
21 subplot(3,3,3), imagesc(fx,fy,Ichapmod.^5), colorbar, axis('square')
22 title('|TF(img)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
23
24 % filtre => fct de transfert
25 h=-ones(3,3); h(2,2)=9; % filtre passe-haut
26 hchap=fft2(h,dim,dim); % fct de transfert corr.
27 hchapmod=abs(fftshift(hchap));
28 subplot(3,3,6), imagesc(fx,fy,hchapmod), colorbar, axis('square')
29 title('|TF(Passe-haut)|'), xlabel('f_x'), ylabel('f_y')
30
31 % filtrage dans l'espace de Fourier
32 Ichapfilt=Ichap.*hchap;
33 Ichapfiltmod=abs(fftshift(Ichapfilt));
34 subplot(3,3,9), imagesc(fx,fy,Ichapfiltmod.^5), colorbar, axis('square')
35 title('|TF(img filt.)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
36
37 % image filtrée
38 Ifilt=real(ifft2(Ichapfilt));
39 subplot(3,3,7), imagesc(Ifilt), colorbar, axis('square')
40 title('img filt.'), xlabel('x'), ylabel('y')
41
42 subplot(3,3,8), plot(Ifilt(64,:),'.'), title('coupe(img filt.)')
43 axis('square'), xlim([2,dim-2]), ylim([-3 3])
```



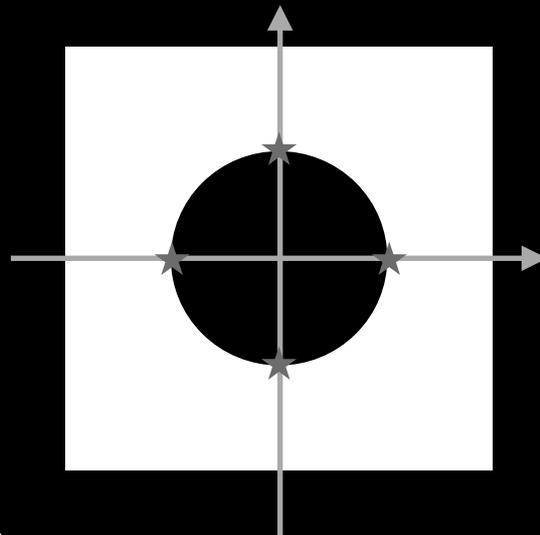
Filtrage passe-haut, passe-bande/coupe-bande, dérivateur — directement dans le plan de Fourier

L'avantage d'être dans le plan de Fourier, c'est que l'on peut modeler les fréquences comme on le désire... Y compris le faire de manière plus simple.

- Cas du filtrage passe-haut : créer \hat{h} tel que $|\hat{h}|$ croît quand la fréquence (spatiale) croît également => on favorise les fréquences les plus élevées => effet contraire du filtre passe-bas.

Application : Atténuer les variations lentes de l'image, pour faire ressortir les variations brusques (contours).

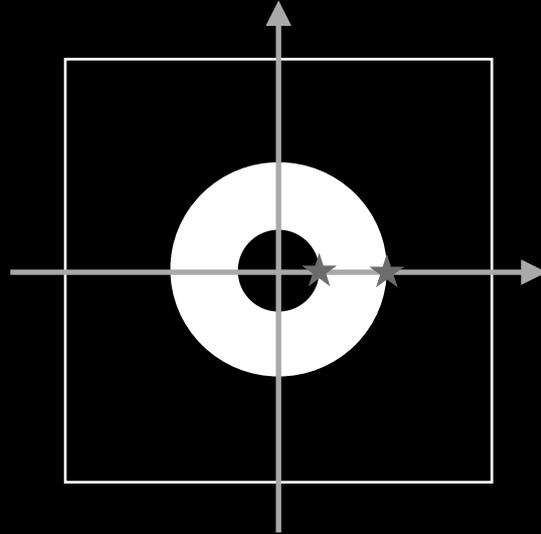
Exemple :



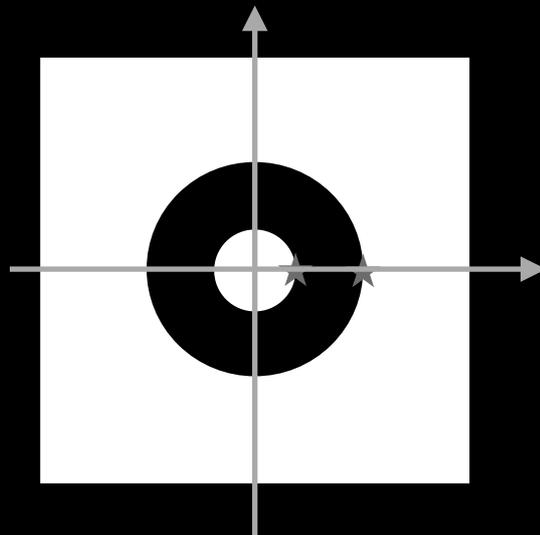
★ fréq. de coupure

(il s'agit bien d'un filtrage directement dans le plan de Fourier, appliqué à la TF de l'image)

- Cas du filtrage passe-bande : on laisse une bande de fréquences entre deux fréquences de coupures.



- Cas du filtrage coupe-bande :



- Filtre dérivateur :

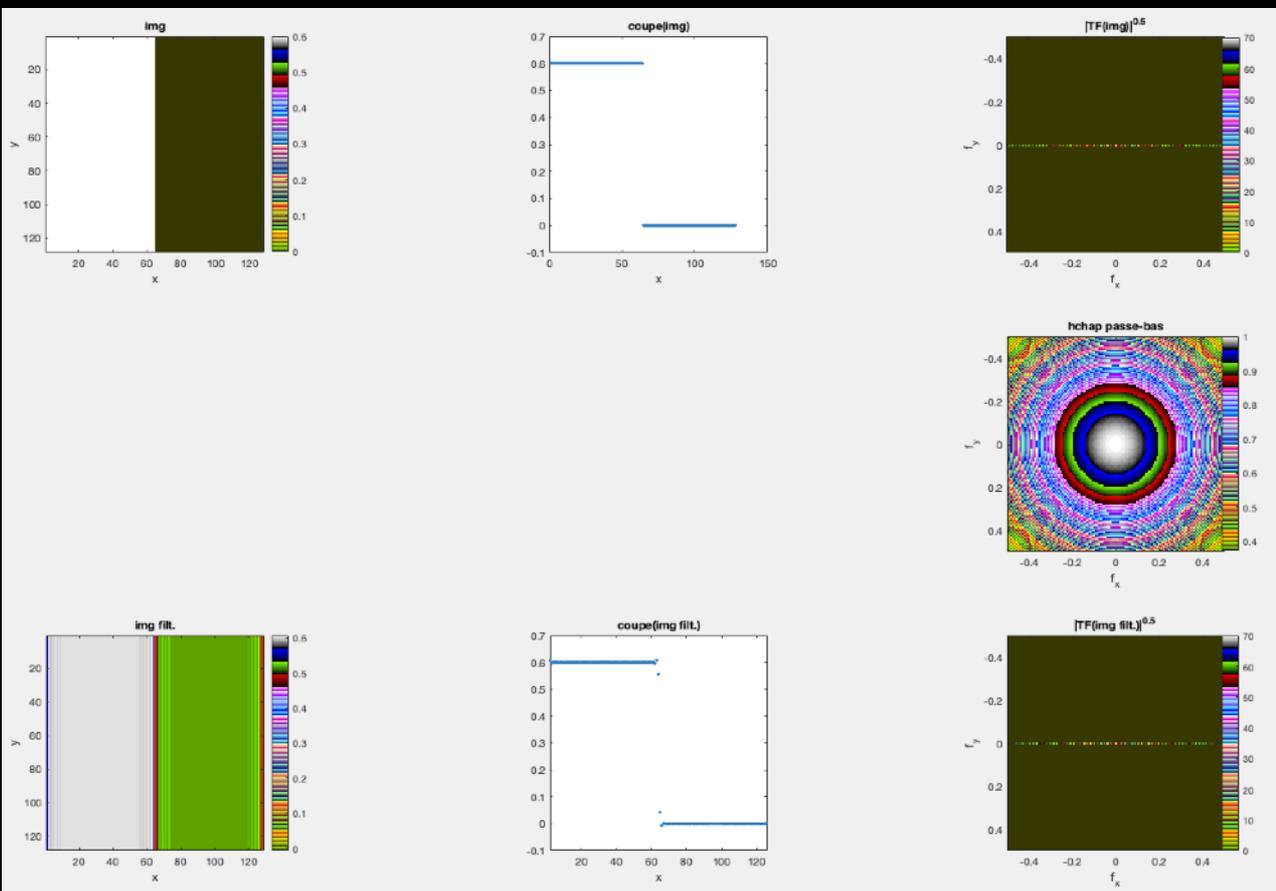
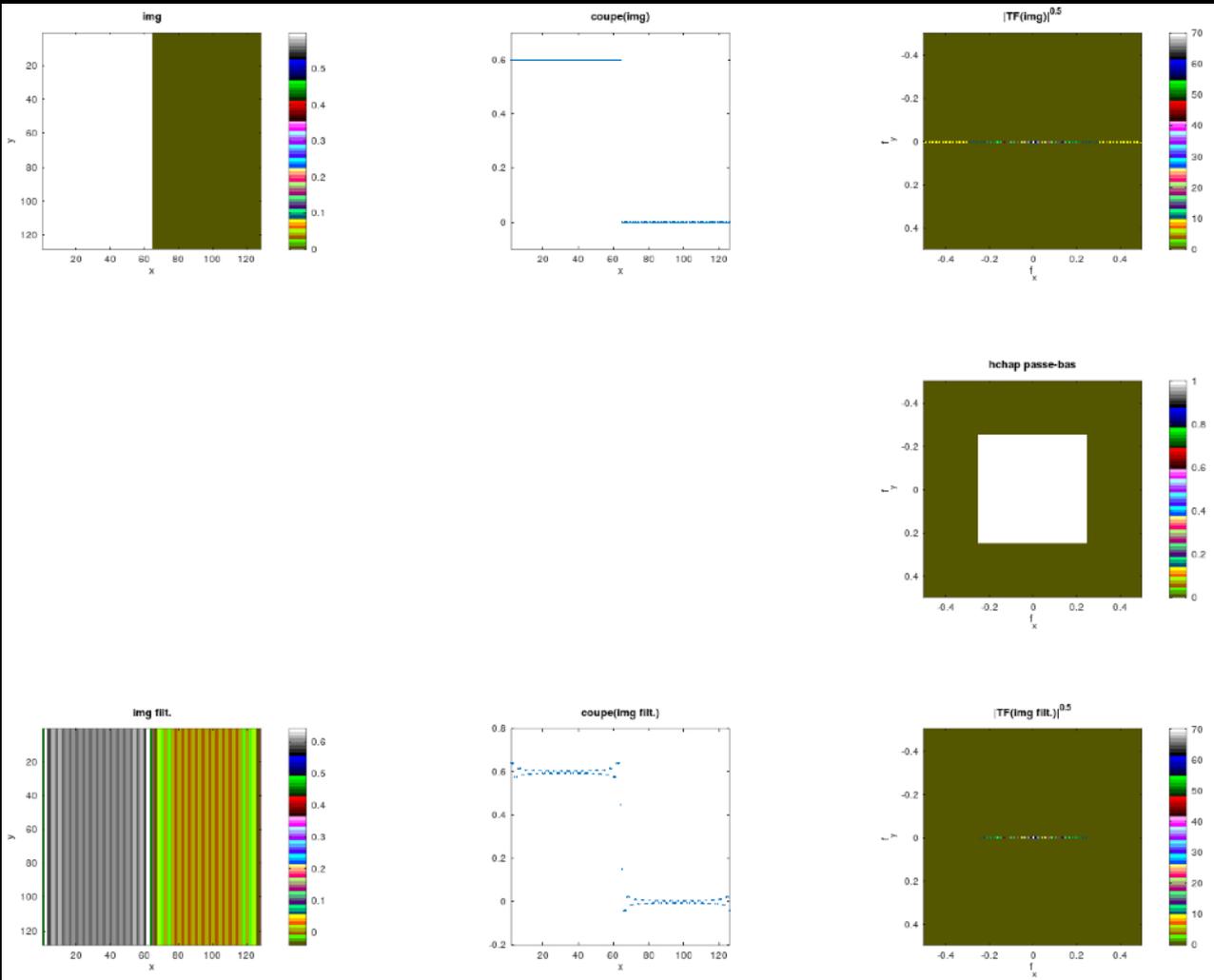
En fait un filtre passe-haut, construit à partir de :

$$f'(x,y) \xrightarrow{TF} 2i\pi u \ 2i\pi v \ \hat{f}(u,v)$$

Exercice 8 :

Filtrer (en masquant dans le plan de Fourier : notion de fonction de transfert) avec un filtre passe-bas, carré, de largeur $dim/2$. Commenter le résultat.

```
1 clear
2 close all
3 %pkg load image
4
5 % image
6 dim=128; I=zeros(dim,dim); I(:,1:dim/2)=0.6; % marche 0.6/0
7 fx=((0:dim-1)-dim/2)/dim; fy=fx; % fréquences
8
9 figure, colormap('colorcube') % colormapp "colorcube"
10
11 subplot(3,3,1), imagesc(I)
12 colorbar, axis('square')
13 title('img'), xlabel('x'), ylabel('y')
14
15 subplot(3,3,2), plot(I(dim/2,:),'.')
16 axis('square'), ylim([-1 .7])
17 title('coupe(img)'), xlabel('x')
18
19 % FFT(image)
20 Ichap=fft2(I,dim,dim);
21 Ichapmod=abs(fftshift(Ichap));
22 subplot(3,3,3), imagesc(fx,fy,Ichapmod.^5)
23 colorbar, axis('square')
24 title('|TF(img)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
25
26 % "filtre" (en fait fct de transfert directement!)
27 hchap=zeros(dim,dim); nn=dim/2;
28 hchap(dim/2-nn/2+1:dim/2+nn/2,dim/2-nn/2+1:dim/2+nn/2)=1.;
29 % plus smooth : le filtre gaussien...
30 % hchap=fspecial('gaussian',dim,nn); hchap=hchap/max(max(hchap));
31
32 subplot(3,3,6), imagesc(fx,fy,hchap)
33 colorbar, axis('square')
34 title('hchap passe-bas'), xlabel('f_x'), ylabel('f_y')
35
36 % filtrage dans l'espace de Fourier
37 Ichapfilt=fftshift(hchap).*Ichap;
38 Ichapfiltmod=abs(fftshift(Ichapfilt));
39 subplot(3,3,9), imagesc(fx,fy,Ichapfiltmod.^5)
40 colorbar, axis('square')
41 title('|TF(img filt.)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
42
43 % retour dans l'espace réel
44 Ifilt=ifft2(Ichapfilt);
45 subplot(3,3,7), imagesc(real(Ifilt))
46 colorbar, axis('square')
47 title('img filt.'), xlabel('x'), ylabel('y')
48
49 subplot(3,3,8), plot(real(Ifilt(64,:)),'.')
50 axis('square'), ylim([-1 .7])
51 title('coupe(img filt.)'), xlim([2,dim-2]), xlabel('x')
```



Exercice 8bis :

Même chose avec un filtre (en fait sa FFT, i.e. un masque) circulaire (10 frequels de rayon)...

```
1 clear
2 close all
3 %pkg load image
4
5 % image
6 dim=128; I=zeros(dim,dim); I(:,1:dim/2)=0.6;
7 fx=((0:dim-1)-dim/2)/dim; fy=fx;
8
9 figure, colormap('colorcube')
10
11 subplot(3,3,1), imagesc(I)
12 colorbar,axis('square')
13 title('img'), xlabel('x'), ylabel('y')
14
15 subplot(3,3,2), plot(I(64,:),'.'), axis('square'), ylim([-0.1 .7])
16 title('coupe(img)'), xlabel('x')
17
18 % FFT(image)
19 Ichap=fft2(I);
20 Ichapmod=abs(fftshift(Ichap));
21 subplot(3,3,3), imagesc(fx,fy,Ichapmod.^5)
22 colorbar, axis('square')
23 title('|TF(img)|'), xlabel('f_x'), ylabel('f_y')
24
25 % "filtre" (en fait fct de transfert directement!)
26 hchap=zeros(dim,dim); rr=10;
27 % hh=fspecial('disk',rr); hh=hh/max(max(hh));
28 % hchap(dim/2+1-rr:dim/2+1+rr,dim/2+1-rr:dim/2+1+rr)=hh;
29 % plus smooth : le filtre gaussien...
30 hchap=fspecial('gaussian',dim,rr); hchap=hchap/max(max(hchap));
31
32
33
34 % filtrage dans l'espace de Fourier
35 Ichapfilt=Ichap.*fftshift(hchap);
36 Ichapfiltmod=abs(fftshift(Ichapfilt));
37 subplot(3,3,9), imagesc(fx,fy,Ichapfiltmod.^5)
38 colorbar, axis('square')
39 title('|TF(img filt.)|^{0.5}'), xlabel('f_x'), ylabel('f_y')
40
41 % retour dans l'espace réel
42 Ifilt=ifft2(Ichapfilt);
43 subplot(3,3,7), imagesc(real(Ifilt))
44 colorbar, axis('square')
45 title('img filt.'), xlabel('x'), ylabel('y')
46
47 subplot(3,3,8), plot(real(Ifilt(64,:)),'.'), axis('square'), ylim([-0.1 .7])
48 title('coupe(img filt.)'), xlabel('x')
```

