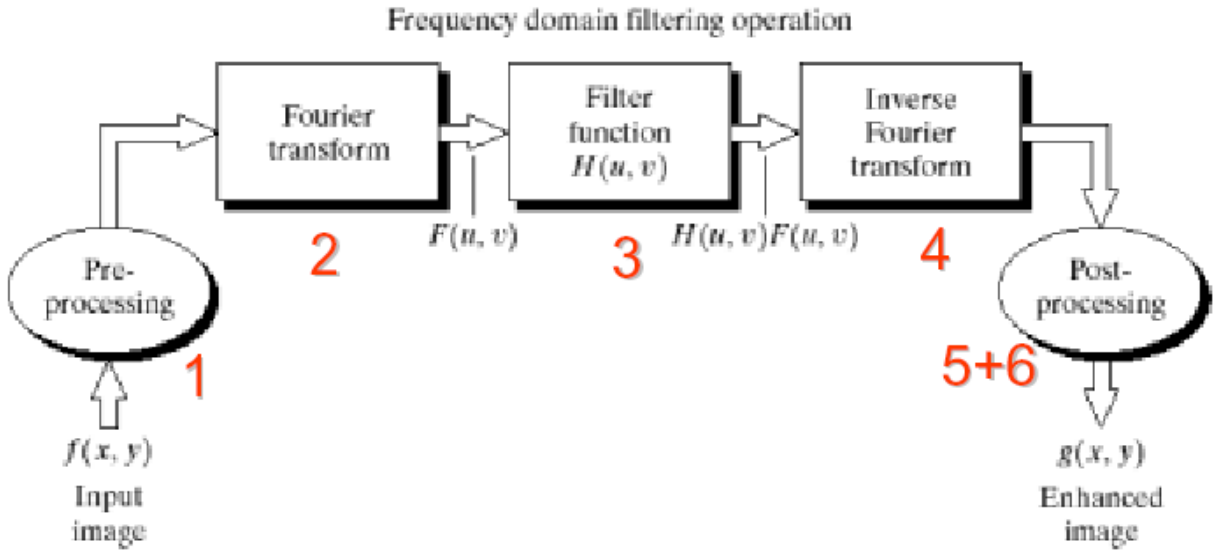
المحاضرة السادسة

تحويل فوريير

لتحويل فوريير دور كبير في تحسين الصورة وتحليلها وتخرينها وضغطها لهذا يعد من أهم التحويلات المستخدمة لمعالجة الصور ذلك بسبب المزايا التي يقدمها لنا نتيجة تحويل الصورة إلى المجال الترددي كالتناظر , عدم تعلقه بالإزاحة وعدم تعلقه بالدوران.....

يمثل الشكل التالي ترتيب العمليات المطبقة عند معالجة الصورة باستخدام تحويل فوريير :



تطبيق تحويل فوريير على نبضة مستطيلة:

```
M = 1000;
f = zeros(1, M);
i = 40;
f(M/2-i:M/2+i) = 1;
Subplot(3,2,1),plot(f);
F = fft(f);
Subplot(3,2,3),plot(F);
Subplot(3,2,4),plot(abs(F));
Fc = fftshift(F);
rFc = real(Fc);
iFc = imag(Fc);
Subplot(3,2,5),plot(abs(Fc));
Subplot(3,2,6),plot(atan(iFc./rFc));
```

المحاضرة السادسة

تطبيق تحويل فوريير على صورة سوداء فيها مربع أبيض:

```
f1 = zeros(500,500);
f1(240:260,230:270) = 1;
subplot(2,2,1);imshow(f1,[]);
F = fft2(f1);
S = abs(F);
subplot(2,2,2);
imshow(S,[]);
Fc = fftshift(F);
S1 = abs(Fc);
subplot(2,2,3); imshow(S1,[]);
S2 = log(1+S1);
subplot(2,2,4);imshow(S2,[]);
```

لكننا في ماتلاب سنقوم بداية بايجاد تحويل فوريير من ثم تطبيق عملية الازاحة من أجل مركزة تحويل فوريير وذلك بعد تحويل قيم الصورة إلى double من ثم نظهر الناتج عن طريق التعليمات التالية:

```
Img=imread('img.jpg');
Img=rgb2gray(Img);
Img=im2double(Img);
F = fft2(Img);
imshow(abs(F),[]);
```

التعليمة fft2 هي التي تمثل تحويل فوريير للصورة المدروسة.
من ثم نقوم باظهار مطال (أي طيف تحويل فوريير) من خلال التعليمة imshow(abs(F),[ ]) والقوسان المتوسطان يسمحان بتمثيل الصورة في التدرجات اللوغاريتمية.

من ثم نطبق الازاحة والتي تكافئ عملية ضرب صورة الدخل بالقيمة $1^{(x+y)}-$ لمركزة تحويلها, عندها سنلاحظ أن الدائرة البيضاء أصبحت في مركز الصورة المعروضة.

```
Fc=fftshift(F);
imshow(abs(Fc),[]);
figure, imshow(log(1+abs(Fc)),[]);
```

يمكن الحصول على الصورة الأصلية بتطبيق فوريير العكسي وذلك بعد القيام بعملية إزاحة عكسية كمايلي:

```
F=ifftshift(Fc);
f=ifft2(F);
imshow(f)
```

ترشيح الصورة في المجال الترددي:

1- مرشحات التمرير المنخفض lpfilter:

يكون التابع من الشكل H = lpfilter(type, M, N, D0, n)

حيث نحدد نوع المرشح من خلال البارامتر type:

1) مرشح التمرير المنخفض المثالي ILPF:

Type :'ideal'
Ideal lowpass filter with cutoff frequency DO. n need not be supplied. DO must be positive.

2) مرشح بتروث للتمرير المنخفض BLPF :

Type : ' btw'
Butterworth lowpass filter of order n , and cutoff DO. The default value for n is 1. DO must be positive.

3) المرشح غوص للتمرير المنخفض GLPF :

Type : 'gaussian'
Gaussian lowpass filter with cutoff (standard deviation) DO. n need not be supplied. DO must be positive.

ثم نحدد حجم هذا المرشح والذي يجب أن يكون مساويا لحجم الصورة المدروسة M-by-N ثم ندخل تردد القطع D0 وقد نحتاج في بعض أنواع المرشحات إلى إدخال درجة المرشح n كما في المرشح butterworth لكن عندما لاندخل قيمة n ستكون مساوية إلى 1 أي n=1 (default).

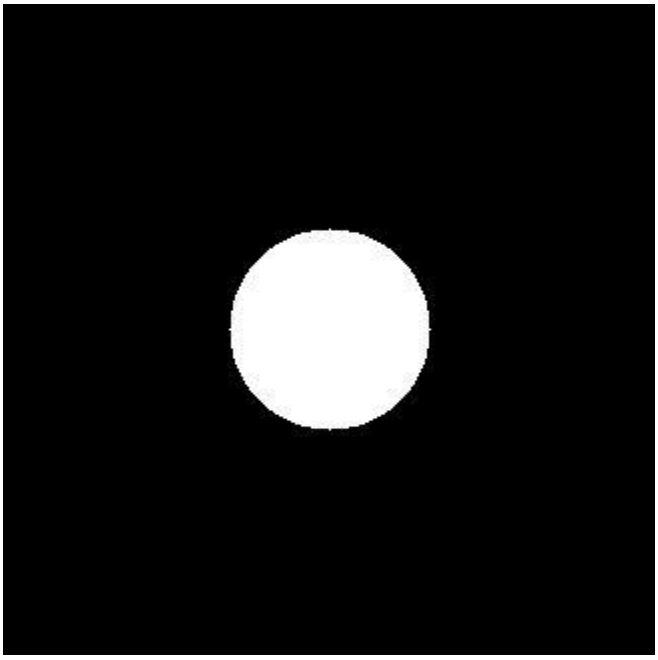
يتم بناء الفلتر بكتابة نوع الفلتر وتحديد حجمه وبارامتراته الهامة من ثم إظهار شكله وذلك كما يلي:

```
[M N]= size (img);

H= lpfilter('ideal',M,N,50);

imshow(fftshift(H),[ ]);
```

المحاضرة السادسة



الشكل السابق يمثل شكل الفلتر عند تردد قطع D0=50

إذا زدنا تردد القطع سيزداد عندها نصف القطر وبالتالي سيزداد عدد الترددات الممكن تمريرها . هذا يعني أننا سنحصل على معلومات أكثر عند تمثيل هذه الصورة.

لقد قمنا ببناء M-file هو عبارة عن تابع اسمه dftfilt(,) يقوم بتطبيق فوريير على الصورة المدخلة ومن ثم يحسب ناتج ضرب مصفوفة المرشح مع الصورة المدخلة ويطبق عليها فوريير العكسي ثم يظهر النتيجة:

```
g=dftfilt(f,H)
```

حيث f هي المصفوفة المعبرة عن الصورة المدروسة بعد تحويلها إلى double

H هي المصفوفة المعبرة عن المرشح المطلوب .

وبالتالي نحصل على ناتج ترشيح الصورة المدروسة في المجال الترددي من خلال التعليمة

Imshow(g);

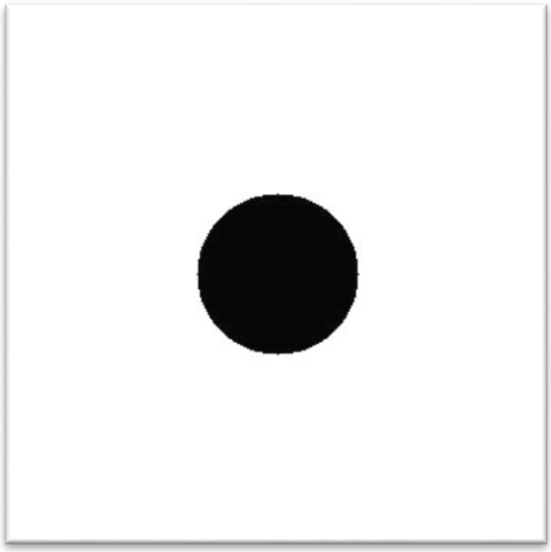
2- مرشحات التمرير المرتفع hpfilter:

هذا التابع له يفس بارامترات تابع مرشح التمرير المنخفض lpfilter

```
H  = hpfilter(type, M, N, DO, n )
```

لكنه يقوم ببناء مرشح للتمرير العالي الذي يكون شكله إذا كان المرشح IHPF كمايلي:

المحاضرة السادسة



أمـا بقيـة الخطـوات فتبقـى كمـاهي وتكـون النتيجـة هـي صـورة ناتجـة عـن تطبيـق مرشـح التمريـر العـالي فـي المجـال الترددي.

المحاضرة السادسة

ملحق بملفات ال M-file المستخدمة في هذه الجلسة:

lpfilter:

```
function [H D]=lpfilter(type,M,N,D0,n)
[U V]=dftuv(M,N);
D=sqrt(U.^2+V.^2);
switch type
    case'ideal'
        H=double(D<=D0);
    case'btw'
        if nargin==4
            n=1;
        end
        H=1./(1+(D./D0).^(2*n));
    case'gaussian'
        H=exp(-(D.^2)./(2*(D0^2)));
    otherwise
        error('unknown filter type')
end
```

dftuv

```
function [U V]=dftuv(M,N)
u=0:(M-1);
v=0:(N-1);
idx=find(u>M/2);
u(idx)=u(idx)-M;
idy=find(v>M/2);
v(idy)=v(idy)-N;
[V U]=meshgrid(v,u);
```

dftfilt

```
function g=dftfilt(f,H)
F=fft2(f,size(H,1),size(H,2));
g=real(ifft2(H.*F));
g=g(1:size(f,1),1:size(f,2));
```

المحاضرة السادسة

hpfilt

```
function [ H , D ] = hpfilter(type, M, N, DO, n )
% Generate highpass f i l t e r .
if nargin ==4
n=1;
end
[H,D] = lpfilter(type, M , N , DO, n);
H = 1 - H;
```

****