Subject: Again an FFT question
Posted by Helder Marchetto on Tue, 26 Jun 2012 21:11:28 GMT
View Forum Message <> Reply to Message

Dear FFTers,
I was just wondering about the translation of the FFT so that the what is located at (0,0) goes in the middle of the image (N/2,M/2).
Until now I did this doing simply a shift of the FFT image, that is:
FFT_Img = SHIFT(FFT(Img),N/2,M/2)
Now I have seen that some people use a FFT "trick" to shift the image. They switch every second pixel of an image to its negative value. This is justified by the translation properties of the DFTs and results in a translation of half the image size (for those seeking to understand the math, try to multiply the function (image) by exp(i*2*Pi(u0*x/N)) and after integration you will get a translation of the Fourier image (signal for 1D) of F(u-u0).

I have tested the difference between the two shifting methods with the following code (in double precision just to be sure I wouldn't get rounding errors or so):

```
;*************************
n = 256
img = RANDOMU(S, n, n) ; Eventually put your image here, with the *right* size
xx = LINDGEN(n) # (LONARR(n)+ 1)
yy = (LONARR(n)+ 1) # LINDGEN(n)
xxyy = xx+yy
OddCenteredImg = Img
EvenCenteredImg = Img
Odd = WHERE((xxyy MOD 2), COMPLEMENT=Even)
OddCenteredImg[Odd] = -Img[Odd]
EvenCenteredImg[Even] = -Img[Even]
WINDOW, XSIZE=2*n,YSIZE=3*n
TVSCL, Img, 0   ;Transform by standard shifting
TVSCL, OddCenteredImg, 1   ;Transform by standard shifting
TVSCL, ALOG(ABS(SHIFT(FFT(Img,/DOUBLE),n/2,n/2))), 2   ;Transform by standard shifting
TVSCL, ALOG(ABS(FFT(OddCenteredImg, /DOUBLE))), 3     ;Transform by pixel sign inversion
TVSCL, ALOG(ABS(SHIFT(FFT(Img,/DOUBLE),n/2,n/2)) - ABS(FFT(OddCenteredImg,
/DOUBLE))), 4  ;Power Spectrum Image Difference
TVSCL, ALOG(ABS(SHIFT(FFT(Img,/DOUBLE),n/2,n/2)) - ABS(FFT(EvenCenteredImg,
/DOUBLE))), 5  ;Power Spectrum Image Difference
;*************************
```

The result is that the two are not exactly the same. Very similar, but not the same.
I have tried varying the size of the image or switching even instead of odd numbers index numbers , but could not get any improvement (with odd image sizes, the difference is even higher).

Does anybody have a reason to use one way (shift(fft(Img...))) rather than the other (switch pixels with index ((x+y) MOD 2 EQ 1)?

I'm more confident using SHIFT, but I would just like to understand why the other method gives different values.

Thanks,
Helder