Subject: Correction: 2D FFt
Posted by Walid on Tue, 22 Oct 1996 07:00:00 GMT
View Forum Message <> Reply to Message

Hi,

I just realized that FFT(data,-1) takes the FFT of an array of up to 7 dimensions. What confused me was that when I plotted, say, the FFT of a gaussian, I got values only near the sides. However, the FFT of a gaussian is a gaussian, so I thought that the FFT routine must not be working correctly for my array. Just in case anyone out there is interested, the problem lies in the way IDL stores the FFT data. The edges are taken as 0 frequency, rather than the more intuitively obvious center to be zero. A simple shift in the FFT'd image solves the problem, and yields the expected results. The code which illustrates this is:

```
x=shift(dist(201),100,100) 'create an array of r-values
z=exp(-(x/10)^2) 'gaussian with spot size (1/e) of 10.
shade_surf,z 'plot the gaussian
Fz=FFT(z,-1) 'take the FFT of z--note that Fz is now of complex type
shade_surf,abs(Fz) 'this will give the wrong picture!
Fzfix=shift(Fz,100,100) 'correct the transform so that the center of the
                        image is at zero frequency
shade_surf,abs(Fzfix) 'the FFT of a gaussian is a gaussian.
IFz=FFT(Fz,1) 'take the inverse fft.
shade_surf,IFz 'gives the original data, as expected.
```

Does anyone know of a more direct (and elegant) way of performing this transform? And why does IDL store the 2D transform this way--doesn't the usual 2D transform treat the center of the array as zero frequency, so as to get a symmetrical function given a symmetrical image?

Hope someone besides myself found this useful!

Walid