Subject: Re: FFT and ROTATE
Posted by Wox on Sat, 06 Sep 2008 11:47:20 GMT
View Forum Message <> Reply to Message

On Fri, 5 Sep 2008 14:04:11 -0700 (PDT), wheinz@gmail.com wrote:

> The fact that the inverse transformations return the original image
> suggests that there is a phase shift introduced somewhere in the
> transform. I can understand that if the array has an even number of
> elements in x or y, then a rotation forces a translation of the
> pixels. But an array with an odd number of elements in x and y should
> not, especially if we are only looking at 90 degree rotations.
>
> Why aren't the sorted lists of the real parts of the coefficients from
> the rotated fft and the fft of the rotated image equal?

Maybe it has something to do with this (see rotation and edge
effects):
http://www.cs.unm.edu/~brayer/vision/fourier.html
There you see that the modulus is also different when you compare
mod(fft(rot(image))) with mod(rot(fft(image))). Apparently the fact
that the moduli were the same in your case, was because you rotated 90
degrees. See below for rotating 45 degrees. Someone with a better
knowledge of fourier transformation can probably explain better. So I
don't think it's a rounding problem, but a problem that comes with
images having a finite size.

```
pro rotFFTtest3
  ;;make image
   image=rebin(bytscl(cos(!pi/18.*findgen(355))),355,355,/sampl e)
  image90 = rot(image,-45)
  i0=52
  i1=i0+250
  image=image[i0:i1,i0:i1]
  image90=image90[i0:i1,i0:i1]

  ;display the images
  window,0
  tvscl,image,0
  tvscl,image90,1
  n = size(image,/dim)
  nfreq=n/2+1 ; # positive freq in each dim
  nfreq_m=nfreq-1-(~(n mod 2)) ; # negative fequencies in each dim

  ;;take fft of image
  f = fft(image)
```

```
  ;;shift it
  f = shift(f,-nfreq[0],-nfreq[1])

  ;;rotate the fft 90 degrees
  f90_1 = rot(f,45)

  ;;shift it back
  f90_1 = shift(f90_1,nfreq[0],nfreq[1])

  ;;take the fft of image90 -- the rotated image
  f90_2 = fft(image90)

  tvscl,fft(f90_1,/inverse),2; fft(rot(fft(image)))
  tvscl,fft(f90_2,/inverse),4; fft(fft(rot(image)))

  ;;phase shift between fft(rot(image)) and rot(fft(image))
  f90_1phase=atan(f90_1,/phase)
  f90_2phase=atan(f90_2,/phase)
  phshift=abs(f90_1phase-f90_2phase)*180/!pi
  f90_1mod=abs(f90_1)
  f90_2mod=abs(f90_2)
  moddiff=abs(f90_1mod-f90_2mod)

  window,1
  tvscl,shift(moddiff,-nfreq[0],-nfreq[1])<0.1,1
  tvscl,shift(phshift,-nfreq[0],-nfreq[1]),0
  print,'fft(rot(image)) vs. rot(fft(image))'
  print,'Phase shift range (degrees): ',min(phshift),max(phshift)
  print,'Modulus diff range: ',min(moddiff),max(moddiff)
end
```