Subject: Re: ROT is ROTTEN
Posted by Richard French on Wed, 21 Nov 2001 06:06:05 GMT

View Forum Message <> Reply to Message

Bhautik Jitendra Joshi wrote:

```
>
> MOO>a=findgen(5,5) & print, a
>      0.00000     1.00000     2.00000     3.00000     4.00000
>      5.00000     6.00000     7.00000     8.00000     9.00000
>      10.0000     11.0000     12.0000     13.0000     14.0000
>      15.0000     16.0000     17.0000     18.0000     19.0000
>      20.0000     21.0000     22.0000     23.0000     24.0000
>
> MOO>print, total(a)
>      300.000

> MOO>print, total(rot(a,360))
>      262.000
>
```

Well, this is interesting! In IDL5.5, I get 262.000 on Windows98 and
Tru64.
In IDL5.4 on Windows, I get 247.000, but 262.000 on Tru64.
Anyone with earlier versions want to weigh in
here? It is hard to see why a rotation of 360 would not give you
what you started with, unless I am really missing something
fundamental...

(I just compared the source code on the 5.4 and 5.5 lib versions of rot
and they are identical except in the comment fields...)

Now if you do:

```
IDL> print,total(rot(dindgen(5,5),360.d0))
      242.00000
IDL> print,!version
{ alpha OSF unix Compaq Tru64 5.5 Aug 28 2001      64      64}

IDL> print,total(rot(dindgen(5,5),360.d0))
      282.00000
IDL> print,!version
{ x86 Win32 Windows Microsoft Windows 5.5 Aug 28 2001      32      64}
```

and here is a nice one... if you put the angle in integer degrees, you
get
one answer, in double precsission, you get another...

```
IDL> print,total(rot(dindgen(5,5),360))
      262.00000
IDL> print,total(rot(dindgen(5,5),360.d0))
```

```
     242.00000
IDL> print,!version
{ alpha OSF unix 5.4 Sep 25 2000      64      64}
```

I modified rot.pro to do things in double precision but it did not make
any difference. I agree that when you turn on /cubic you don't get
300.00, but something close to it. I have not displayed the images
to see what they look like, but it surely is not doing what we think
it should!

Dick
French

Subject: Re: ROT is ROTTEN
Posted by Wayne Landsman on Wed, 21 Nov 2001 06:32:34 GMT

View Forum Message <> Reply to Message

>
> ROT is bad. Can it be fixed? Is there a (fast) alternative?

The easiest alternative is to use the intrinsic ROTATE function, which is specifically designed to deal with rotations of multiples of 90 degress.

A more subtle alternative is to add the MISSING keyword when using the ROT() function, e.g.

```
print, total(rot(a,270,/interp, MISSING = !VALUES.F_NAN))
```

and you will find that there are values flagged as "missing".   This is because unless the rotation is exactly a multiple of 90 degrees, then there will be subpixels in the output array for which there are no corresponding values to interpolate in the input array.   Instead of extrapolating, these pixels get flagged as "missing" and the values returned by ROT() are not to be trusted.   (Exactly which pixels get flagged as missing depends on the roundoff error.)

This question has come up before here -- perhaps RSI should modfiy ROT() so that it calls ROTATE when the user supplies an exact integer mulitple of 90 degrees.

Wayne Landsman   landsman@mpb.gsfc.nasa.gov

P.S. I would also like a keyword to ROT() to control the output dimensions
e.g. ftp://idlastro.gsfc.nasa.gov/landsman/idl/rot.pro