## Subject: Re: How to define resolution (not dimension!) for IDLanRoi->computeMask Posted by justspam03 on Tue, 10 Feb 2009 12:00:54 GMT

View Forum Message <> Reply to Message

Thanks. I've been hacking a bit and it seems in the end I'll extract the data points

from the ROI, scale them properly, create a 'helper' roi and let the latter do the

computeMask. The advantage is a.o. that this way I can also non-square pixels (which

is a requirement).

PolyFillv might not be first choice as it implicitly maps the bounding box of the 'Roi'

to an integer number of pixels (if I understood that correctly.)

Still, if I may ask again: is anyone aware of the difference between the 'location' and the 'pixel center' keyword of IDLanROI->computeMask?

Cheers Oliver

On Feb 9, 4:11 pm, David Fanning <n...@dfanning.com> wrote:

> justspa...@yahoo.de writes:

>> eieieiei - I was afraid I'd need to do something like this :-(

>> Well, I guess I'll have my own overloaded computeMask... I anyway

>> subclass from IDLanRoi.

>

Well, somehow you have to get from a data coordinate system

> into a pixel coordinate system. There are probably other

> ways to do it. I showed you what I do in direct graphics.

> There are certainly comparable methods in object graphics,

> but I don't know what they are. :-)

> I suppose you could use the old Scale\_Vector/Value\_Locate

> trick to find the device values of your ROI:
>
> s = Size(image)
> xvec = Scale\_Vector(Findgen(s[0], 1, 5)
> yvec = Scale\_Vector(Findgen(s[1], 1, 5)
> xdevice = Value\_Locate(xvec, x)
> ydevice = Value\_Locate(yvec, y)

```
indices = PolyFillv(xdevice, ydevice, s[0], s[1])
I haven't tested that, but it should work. :-)
Cheers,
David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming:http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")
```