Subject: Re: polyfillv and the boundary pixels
Posted by David Fanning on Thu, 15 Jan 2004 20:53:33 GMT
View Forum Message <> Reply to Message

Bruce Bowler writes:

> Any options I missed in polyfillv? Any routine I missed that will give
> just the border? Any other thoughts on how to meet my goals?

Have you tried IDLanROI? I've run into this problem before,
and tried a number of things. (Didn't I write an article about
this!?) Anyway, let me know if you have any luck with this,
otherwise I'll look around some. Have to go...

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: polyfillv and the boundary pixels
Posted by Karsten Rodenacker on Fri, 16 Jan 2004 08:02:46 GMT
View Forum Message <> Reply to Message

Search in the list (http://groups.google.com/) for "Area of a blob" and
"object IDLanROI and polyfillv". I think the point mentioned is still an
open topic.

Regards

Bruce Bowler wrote:

> Situation, I want to find the pixels within AND on the boundry of a
> polygon defined by an array of vertices. polyfillv does the "inside" and
> some of the pixels on the border (see the reference guide for a brief
> description of how they decide and a pointer to a more exhaustive treatise
> on the subject).
>
> Simplistic example
>
> array = fltarr(30,30)
> x = [19,20,20,19]
> y = [19,19,20,20]
> z = polyfillv(x,y,30,30)
>

> z ends up with only 1 pixel in it, 589, which is (if I did the math right
> :-) [19,19]. What I'd like to find in z is [589,590,619,620].
>
> Obviously, my polygons are a little more complex than the above, some with
> 100 or more vertices.
>
> Any options I missed in polyfillv? Any routine I missed that will give
> just the border? Any other thoughts on how to meet my goals?
>
> IDL 5.6, in case it matters.
>
> Thanks!
> Bruce
>

Subject: Re: polyfillv and the boundary pixels
Posted by JD Smith on Sat, 17 Jan 2004 16:36:09 GMT
View Forum Message <> Reply to Message

On Thu, 15 Jan 2004 13:39:13 -0700, Bruce Bowler wrote:

> Situation, I want to find the pixels within AND on the boundry of a
> polygon defined by an array of vertices. polyfillv does the "inside"
> and some of the pixels on the border (see the reference guide for a
> brief description of how they decide and a pointer to a more exhaustive
> treatise on the subject).
>
> Simplistic example
>
> array = fltarr(30,30)
> x = [19,20,20,19]
> y = [19,19,20,20]
> z = polyfillv(x,y,30,30)
>
> z ends up with only 1 pixel in it, 589, which is (if I did the math
> right :-) [19,19]. What I'd like to find in z is [589,590,619,620].
>
> Obviously, my polygons are a little more complex than the above, some
> with 100 or more vertices.
>
> Any options I missed in polyfillv? Any routine I missed that will give
> just the border? Any other thoughts on how to meet my goals?
>
> IDL 5.6, in case it matters.

This is called "clipping", and is used quite a bit in computer
graphics. It's also related to the "anti-aliasing" of fonts. For

scientific applications, read the article on dfanning.com regarding "drizzling". There are a variety of polygon clippers with different strengths and weaknesses (e.g. only convex simple polygons, etc.), but one of the simplest is called the "Sutherland-Hodgeman" clipping algorithm, and can be explained in 3 simple (and easy to find) steps. Unfortunately, this type of algorithm is difficult to optimize in IDL, and my best efforts have led to poorly-performing implementations. I have resorted to calling C code, which still isn't as speedy as I'd like, thanks to CALL_EXTERNAL's overhead. I have put in a feature request at RSI for a vectorized native polygon clipper, with usage similar to POLYFILLV, but which can clip any number of polygons at once. I also requested a vectorized clipper for clipping arbitrary polygons against eachother. If you're interested, let RSI know and maybe we'll see it in the next release.

JD
