Subject: Re: Average over odd/even lines
Posted by jschwab@gmail.com on Thu, 07 Dec 2006 04:54:58 GMT
View Forum Message <> Reply to Message

One caveat is that I'm not exactly sure how you want to deal with the last even numbered line. But here's what I would probably do. (I'm sure JD, David, or someone else can probably come up with something a bit more clever.)

```
; picks off the odd rows
oddlines = oldimage[*, indgen(240) * 2]

; averages odd row with odd row below it by adding the odd lines
; to themselves shifted up a row
evenlines = (oddlines + shift(oddlines, -640)) * .5

; puts the even rows after the odd rows
; and then reshapes so they are below
new_image = reform([oddlines, evenlines], 640, 480)
```

The shift statement moves row 3 -> row 1, and row 1 -> row 479, that means in the final image, row 480 is the mean of row 1 and row 479, which is probably not how you want it.

Other than the last row, I think this is a decent way, though it doesn't use histogram. :-)

Cheers,
Josiah

Pete wrote:
> Hi All,
>
> I am trying to write an IDL program for "smoothing" over lines of image
> data acquired with an aerial CCD system. This requires reading the odd
> lines, calculating the mean and placing it in the even. The images are
> a constant 640x480 pixels.
>
> i.e.)
> line 1 :   2 2 2 2...
> line 2 :   x x x x...
> line 3 :   4 4 4 4...
>
> After processing,
>
> line 1 :   2 2 2 2...

> line 2 :   3 3 3 3...
> line 3 :   4 4 4 4...
>
> I can think of several ways to implement this but I thought the group
> may point me to the most efficient.
>
> Thanks,
> Pete

Subject: Re: Average over odd/even lines
Posted by peter.eddy@shaw.ca on Thu, 07 Dec 2006 18:45:09 GMT

View Forum Message <> Reply to Message

Thanks Josiah,
This does work well, better than my original program with loops.
The last line will likely be clipped or maybe I can pad the image prior
to processing.

At any rate thanks for your suggestion,
Pete

Subject: Re: Average over odd/even lines
Posted by JD Smith on Thu, 07 Dec 2006 18:58:50 GMT

View Forum Message <> Reply to Message

On Wed, 06 Dec 2006 19:46:16 -0800, Pete wrote:

> Hi All,
>
> I am trying to write an IDL program for "smoothing" over lines of image
> data acquired with an aerial CCD system. This requires reading the odd
> lines, calculating the mean and placing it in the even. The images are
> a constant 640x480 pixels.
>
> i.e.)
> line 1 :   2 2 2 2...
> line 2 :   x x x x...
> line 3 :   4 4 4 4...
>
> After processing,
>
> line 1 :   2 2 2 2...
> line 2 :   3 3 3 3...
> line 3 :   4 4 4 4...
>

> I can think of several ways to implement this but I thought the group
> may point me to the most efficient.

This is a good chance to use the mostly neglected stride operator for
IDL's range subscripts, which has the syntax [low:high:stride].

```
 d=size(a,/DIMENSIONS)
 x=indgen(d[0]) & y=findgen((d[1]-1)/2)+.5
 a[*,1:d[1]-2:2]=interpolate(a[*,0:*:2],x,y,/GRID)
```

Note that the last line is unchanged for images with an even number of
lines (that comes from (d[1]-1)/2).

Since your dimensions never change, you can cache x & y and use them
over and over. For this reason as well, it might also be faster to
expand out [*,1:d[1]-2:2] etc. into index arrays and cache them,
rather than have IDL recompute them for each image. It would be
lovely if IDL provided a function to convert a given subscript syntax
into an array of indices, but I don't believe it has one.

JD

Subject: Re: Average over odd/even lines
Posted by peter.eddy@shaw.ca on Fri, 08 Dec 2006 20:39:47 GMT

View Forum Message <> Reply to Message

Cool, I had never heard of the stride operator.
I will likely keep the code as you wrote it just in case the program is
used with another system with different dimentions.

Thanks for your help!
Pete