
Subject: Re: Interpolation question

Posted by [Craig Markwardt](#) on Wed, 25 Jul 2001 19:27:16 GMT

[View Forum Message](#) <> [Reply to Message](#)

rkj@dukebar.crml.uab.edu (R. Kyle Justice) writes:

> I have an image containing some bad values. I would like
> to replace these points with the average value of their
> neighbors. Is there an easy way to do this without loops?
>
> For instance, a 3x3 array as follows:
>
> 1 1 1
> 1 0 1
> 1 1 1
>
> would become
>
> 1 1 1
> 1 1 1
> 1 1 1.
>
> I can't get the boundary conditions to work correctly when
> I use CONVOL. It either zeros the edges or does not process
> them.

How about a hybrid approach? Use CONVOL for the center of the image,
and then special-case the stripes around the edges. Or, better yet,
check out the MEDIAN function for sliding median smoothing.

Good luck,
Craig

--

Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response

Subject: Re: Interpolation question

Posted by [thompson](#) on Thu, 26 Jul 2001 15:59:39 GMT

[View Forum Message](#) <> [Reply to Message](#)

rkj@dukebar.crml.uab.edu (R. Kyle Justice) writes:

> I have an image containing some bad values. I would like
> to replace these points with the average value of their

> neighbors. Is there an easy way to do this without loops?

> For instance, a 3x3 array as follows:

```
> 1 1 1
> 1 0 1
> 1 1 1
```

> would become

```
> 1 1 1
> 1 1 1
> 1 1 1.
```

One way to do this would be something like the following.

```
SZ = SIZE(ARRAY)
NX = SZ(1)
NY = SZ(2)
GOOD = ARRAY EQ ARRAY ;Bad points are NaN
TEMP = ARRAY*GOOD
WBAD = WHERE(GOOD EQ 0, COUNT)
IF COUNT GT 0 THEN BEGIN
  I = WBAD MOD NX
  J = WBAD / NX
  ARRAY[I,J] = (TEMP[I-1,J-1] + TEMP[I,J-1] + TEMP[I+1,J-1] + $
    TEMP[I-1,J] + TEMP[I+1,J] + TEMP[I-1,J+1] + TEMP[I,J+1] + $
    TEMP[I+1,J+1]) / (GOOD[I-1,J-1] + GOOD[I,J-1] + $
    GOOD[I+1,J-1] + GOOD[I-1,J] + GOOD[I+1,J] + GOOD[I-1,J+1] + $
    GOOD[I,J+1] + GOOD[I+1,J+1])
ENDIF
```

> I can't get the boundary conditions to work correctly when
> I use CONVOL. It either zeros the edges or does not process
> them.

This doesn't really trip the boundary quite right either, because some points will be more heavily weighted than others. But it does get the job done. You can modify it with more WHERE() statements, to treat the points at the edges separately.

William Thompson
