Subject: Re: iterating POLY_2D
Posted by David Williams on Mon, 21 Feb 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Just as a postscript...

Christophe Morisset helpfully suggested that I also post the routine that's giving me bother. It's included below. I'd really appreciate any help anyone can give.

Dave Williams.

--

```
;+
;NAME:
;    frames_move.pro
;PURPOSE:
;    To remove the shaking from a sequence of digital
;    images in the form of a 50x40x200 floating point
;    array.
;INPUTS:
;    DATA - the input array
;OUTPUTS:
;    DATAOUT - the stabilised sequence
;    DISPARR - the 2x200 array containing the
;        displacement co-ordinates relative to
;        DATA(*,*,0).
;-
PRO frames_move,data,dataout,disparr

sizD=SIZE(data)
disparr=fltarr(2,sizD(3))
dataout=fltarr(sizD(1),sizD(2),sizD(3))
dataout(*,*,0)=data(*,*,0)
;the original image will be the same in the output
;as it's the reference point from which
;the displacement in the rest of the images
;is measured.

;make the temporary gradient array to
;define the edges of the main feature:
gradex=fltarr(sizD(1),sizD(2),sizD(3))
FOR g=0,sizD(3)-1 DO BEGIN
   gradex(*,*,g)=grad(data(*,*,g))
   ;(I've included this GRAD function
   ;below in a commented section. It
   ;comes from the IDL User's Guide.)
```

```
ENDFOR

;set up the comparison array BIGJUMP and the
;contents of the first image equal to
;the original image in the sequence passed.
bigjump=fltarr(sizD(1)-1,sizD(2)-1,2)
bigjump(*,*,0)=gradex(1:sizD(1)-1,1:sizD(2)-1,0)

FOR k=1,sizD(3)-1 DO BEGIN
   bigjump(*,*,1)=gradex(1:sizD(1)-1,1:sizD(2)-1,k)
   ;set the second image equal to the image
   ;that is to be compared to the original image,
   ;removing the first X and Y lines which get
   ;wrapped around by the GRAD routine.

   dxy=tr_get_disp(bigjump)
   ;dxy is created by a specialised routine
   ;written for the TRACE SolarSoft IDL distribution
   disparr(0,k)=-dxy(0,1)
   disparr(1,k)=-dxy(1,1)
   ;get the cross-correlation displacement

   ;this is the bit I'm having trouble iterating successfully:
   dataout(*,*,1)=POLY_2D(data(*,*,1),$
   [disparr(0,k),0.,1.,0.],$
   [disparr(1,k),1.,0.,0.],cubic=-0.5)
   PRINT,'polynomial interpolation completed for iteration
'+ARR2STR(k,/trim)

ENDFOR

END

;the GRAD.PRO function is as follows, and is used in FRAMES_MOVE.PRO
;to enhance the accuracy of the routine by detectin the edges of the
;slightly fuzzy main feature:
;
;FUNCTION GRAD, IMAGE
;
;RETURN, ABS(IMAGE - SHIFT(IMAGE,1,0)) + $
;    ABS(IMAGE-SHIFT(IMAGE,0,1))
;
;END
;
```


 == =
David R. Williams, Tel.: (+44 1232) 273509
APS Division,

Pure & Applied Physics Dept.,
Queen's University,
Belfast,
BT7 1NN. http://star.pst.qub.ac.uk/~drw/
 ========== =