
Subject: Re: Can this be done using array operations instead?

Posted by [Jean H.](#) on Thu, 02 Apr 2009 18:39:57 GMT

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

Hi Robin,

ok, so basically you are computing the sum and the square root of your 3*3 images. One option can be, if your computer memory permits it, to create copies of your image, and to shift them...

```
data
data1 = shift(data,1)
data2 = shift(data,1,1)
data3 = shift(data,1,-1)
data4 = shift(data,0,1)
data5 = shift(data,0,-1)
data6 = shift(data,-1)
data7 = shift(data,-1,1)
data8 = shift(data,-1,-1)
```

then you can do whatever you want...

```
sum = data + data1 + ...+ data8
sqrt = sqrt(sum)
```

You get the result for every pixel.

Be careful on the edges, again.

Now this will be a problem if your image is a big one.

Jean

robintw wrote:

```
> Hi Allan,
>
> Thanks for the very informative reply.
>
> The sort of technique that you suggest at the end looks like the sort
> of thing I want to do - but I'm not entirely sure how to go about it.
>
> I've looked at my code again and realised that many values in my
> formula are constant (yes I know some of them can be taken out of the
> loop in the following code - but once I realised I might be able to do
> it with array functions I decided not to try and improve that code any
> more until I'd converted it to use array functions), and in fact only
> a few things vary. What I need to do is step through each 3x3 square
> in the array (not sure exactly what to do about the edges) and get the
```

```

> total for each of those squares, as well as the number of values in
> the square (normally 9 obviously, but if it's at the edge then there
> might be less). I then need to run a calculation which includes these
> total and number, as well as some other constant values.
>
> I've posted my code below, but I'd prefer it if you could describe in
> general how to do these kind of things with some useful examples,
> rather than just taking my code and writing the new version for me.
> This code is part of a project I'm doing for university, and, although
> the code is not the main part of the project (the code is just to help
> me implement a new technique for image processing) I'd rather write
> the code mostly myself.
>
> Here is the code:
>
> ; Creates a Getis image given a FID, the dimensions of the file, a
> distance to use for the getis routine
> ; and a base window to send progress updates to
> PRO CREATE_GETIS_IMAGE, file, dims, distance, report_base
>
> ; Print debugging info
> print, "Distance", distance
>
> ; TODO: Get this to loop through bands
> ; Get the data for the first band of the file (ignores pos from
> earlier)
> WholeBand = ENVI_GET_DATA(fid=file, dims=dims, pos=0)
>
> ; Calculate the dimensions of WholeBand
> SizeInfo = SIZE(WholeBand, /DIMENSIONS)
> NumRows = SizeInfo[0]
> NumCols = SizeInfo[1]
>
> ; Let the progress bar know what the denominator of the fraction is
> - the max
> ENVI_REPORT_INC, report_base, NumRows
>
> ; --- Calculate variable values for the WholeBand
>
> ; Get the global mean
> GlobMean = MEAN(WholeBand)
>
> ; Get the global variance
> GlobVariance = VARIANCE(WholeBand)
>
> ; Get the number of values in the whole image
> GlobNumber = NumRows * NumCols
>

```

```

> ; Create the output array to store the Getis values in - NB: Must be
> an array of floats
>   OutputArray = FLTARR(NumRows, NumCols)
>
> ; For each pixel in the image
>   FOR Rows = 0, NumRows - 1 DO BEGIN
>
>     ; Send an update to the progress window telling it to let us know
>     if cancel has been pressed
>       ENVI_REPORT_STAT, report_base, Rows, NumRows - 1, cancel=cancelled
>
>     ; If cancel has been pressed then...
>     IF cancelled EQ 1 THEN BEGIN
>       ; Close the progress window
>       ENVI_REPORT_INIT,base=report_base, /FINISH
>       ; Exit the function
>       RETURN
>     ENDIF
>     FOR Cols = 0, NumCols - 1 DO BEGIN
>       ; Make sure RowBottom doesn't go below 0
>       RowBottom = Rows - Distance
>       IF RowBottom LT 0 THEN RowBottom = 0
>
>       ; Make sure RowTop doesn't go above NumRows
>       RowTop = Rows + Distance
>       IF RowTop GE NumRows THEN RowTop = NumRows - 1
>
>       ColBottom = Cols - Distance
>       IF ColBottom LT 0 THEN ColBottom = 0
>
>       ColTop = Cols + Distance
>       IF ColTop GE NumCols THEN ColTop = (NumCols - 1)
>
>       ; Get the subset of the image corresponding to the Area of
>       Interest (AOI)
>       AOI = WholeBand[RowBottom:RowTop, ColBottom:ColTop]
>
>       ; Calculate the getis value for this AOI
>       getis = CALCULATE_GETIS(GlobMean, GlobVariance, GlobNumber, AOI)
>
>       ; Set the pixel in the output image equal to the getis value
>       OutputArray[Rows, Cols] = getis
>     ENDFOR
>   ENDFOR
>
> ; Code to scale 0-255 - not used at the moment
>
> ;MaxOutputArray = MAX(OutputArray)

```

```

> ;MinOutputArray = MIN(OutputArray)
> ;RangeOutputArray = MaxOutputArray - MinOutputArray
>
> ;print, MaxOutputArray
> ;print, MinOutputArray
> ;print, RangeOutputArray
>
> ;ScaledArray = OutputArray - MinOutputArray
> ;ScaledArray = ScaledArray * (255 / MaxOutputArray - MinOutputArray)
>
> ; Write the data to an image in ENVI memory
> ENVI_ENTER_DATA, OutputArray
>
> ; Close the progress window
> ENVI_REPORT_INIT,base=report_base, /FINISH
> END
>
> ; Calculates the getis value for an AOI given the AOI as an array, and
> various
> ; values of global constants - Mean, Variance and Number of pixels
> FUNCTION CALCULATE_GETIS, GlobMean, GlobVariance, GlobNumber, AOI
> ; --- Calculate variable values for the AOI
>
> ; Get the Sum of the values in the AOI
> AOISum = TOTAL(aoi)
>
> ; Get number of values in AOI
> SizeInfo = SIZE(aoi, /DIMENSIONS)
> SizeOfSize = SIZE(SizeInfo, /DIMENSIONS)
> IF SizeOfSize EQ 2 THEN AOINumber = SizeInfo[0] * SizeInfo[1]
> IF SizeOfSize EQ 1 THEN AOINumber = SizeInfo[0]
>
> ; --- Start Calculating Getis Statistic
>
> ; Calculate the top of the fraction
> TopFraction = AOISum - (FLOAT(AOINumber) * GlobMean)
>
> ; Calculate the square root
> SquareRootAnswer = SQRT((FLOAT(AOINumber) * (GlobNumber -
> AOINumber))/(GlobNumber - 1))
>
> ; Calculate bottom of fraction
> BottomFraction = GlobVariance * SquareRootAnswer
>
> ; Calculate Getis Statistic
> Getis = TopFraction / BottomFraction
>
> RETURN, Getis

```

> END
>
> Thanks,
>
> Robin
