Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by jeanh on Mon, 08 Mar 2010 12:26:57 GMT
View Forum Message <> Reply to Message

Let's see last point in your message first.
For images, you don't need more than 3 or 4 dimensions.
1) X
2) Y
3) Z - number of images
4) (RGB)

Now, and this may not be the most efficient way but it will allow you to understand what is going on, you can do something like:

```
image1Array
image2Array
nbImagesFirstGroup = 10
nbImagesSecondGroup =  5

for img1 = 0, nbImagesFirstGroup -1 do begin
 for img2 = 0, nbImagesSecondGroup -1 do begin
  result = image1Array[*,*,img1] - image2Array[*,*,img2]
 endfor
endfor
```

here, image1Array has 3 dimensions (x, y, z). By using the subscript [*,*,img1], you are accessing "each elements of the 1 and 2nd dimensions, corresponding to img1", img1 being a simple counter. Of course, in this example, the result will be overwritten in each iteration so you may want to save it.

This being said, you should first write a more precise question. "offset the elements by 1 to the right" can mean lots of things... are yo comparing 2 images, shifting in the X direction? Y? or are you considering multiple images?

Jean

On 08/03/2010 6:43 AM, Will wrote:
> Hi
>
> sorry I don't think the heading was very clear there.
>
> I have loaded a group of arrays (jpeg files) into a seperate float
> array, and again with another group of files I have done the same
> thing. I am now trying to have the two arrays perform a subtraction
> calculation with one another and what I want to do is have the arrays

> do this with each other until they have did the calculation with every
> element. i.e
>
> it does the first calculation, then offsets the elements by one to the
> right and does the next calculation then repeats this until it has
> done every element in the array.
>
> I was curious as to how to do this successflly I am currently trying
> to use a FOR loop after my Repeat loop failed. I have the maths I need
> here but I don't know how to access the subscripts of my array. I have
> inputted all the files into an array using the FLOAT command. the
> likes of FLTARR just keeps saying that I have more than 8 dimensions
> and hence it won't work.
>
> Any ideas guys?
>
> Thanks
>
> Will

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by Spon on Mon, 08 Mar 2010 12:31:08 GMT
View Forum Message <> Reply to Message

On Mar 8, 11:43 am, Will <theloneguitar...@hotmail.co.uk> wrote:
> Hi
>
> sorry I don't think the heading was very clear there.
>
> I have loaded a group of arrays (jpeg files) into a seperate float
> array, and again with another group of files I have done the same
> thing. I am now trying to have the two arrays perform a subtraction
> calculation with one another and what I want to do is have the arrays
> do this with each other until they have did the calculation with every
> element. i.e
>
> it does the first calculation, then offsets the elements by one to the
> right and does the next calculation then repeats this until it has
> done every element in the array.
>
> I was curious as to how to do this successflly I am currently trying
> to use a FOR loop after my Repeat loop failed. I have the maths I need
> here but I don't know how to access the subscripts of my array. I have
> inputted all the files into an array using the FLOAT command. the
> likes of FLTARR just keeps saying that I have more than 8 dimensions
> and hence it won't work.
>

> Any ideas guys?
>
> Thanks
>
> Will

Hi Will,

do you mean like this?:

```
arr1 = [3, 4, 2, 8]
arr2 = [1, 0, 4, 8]
diff = arr1 - arr2

print,diff
      2       4      -2       0
```

'diff' will be a 4-element, one dimensional array in this case, but
can just as easily be multidimensional.
So long as the number of elements in both arrays are the same, simply
using the '-' operator will do.

Regards,
Chris

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by Will on Mon, 08 Mar 2010 12:40:32 GMT
View Forum Message <> Reply to Message

On Mar 8, 12:26 pm, jeanh
<jghasb...@DELETETHIS.environmentalmodelers.ANDTHIS.com> wrote:
> Let's see last point in your message first.
> For images, you don't need more than 3 or 4 dimensions.
> 1) X
> 2) Y
> 3) Z - number of images
> 4) (RGB)
>
> Now, and this may not be the most efficient way but it will allow you to
> understand what is going on, you can do something like:
>
> image1Array
> image2Array
> nbImagesFirstGroup = 10
> nbImagesSecondGroup = 5
>
> for img1 = 0, nbImagesFirstGroup -1 do begin

> for img2 = 0, nbImagesSecondGroup -1 do begin
> result = image1Array[*,*,img1] - image2Array[*,*,img2]
> endfor
> endfor
>
> here, image1Array has 3 dimensions (x, y, z). By using the subscript
> [*,*,img1], you are accessing "each elements of the 1 and 2nd
> dimensions, corresponding to img1", img1 being a simple counter.
> Of course, in this example, the result will be overwritten in each
> iteration so you may want to save it.
>
> This being said, you should first write a more precise question. "offset
> the elements by 1 to the right" can mean lots of things... are yo
> comparing 2 images, shifting in the X direction? Y? or are you
> considering multiple images?
>
> Jean
>
> On 08/03/2010 6:43 AM, Will wrote:
>
>
>
>> Hi
>
>> sorry I don't think the heading was very clear there.
>
>> I have loaded a group of arrays (jpeg files) into a seperate float
>> array, and again with another group of files I have done the same
>> thing. I am now trying to have the two arrays perform a subtraction
>> calculation with one another and what I want to do is have the arrays
>> do this with each other until they have did the calculation with every
>> element. i.e
>
>> it does the first calculation, then offsets the elements by one to the
>> right and does the next calculation then repeats this until it has
>> done every element in the array.
>
>> I was curious as to how to do this successflly I am currently trying
>> to use a FOR loop after my Repeat loop failed. I have the maths I need
>> here but I don't know how to access the subscripts of my array. I have
>> inputted all the files into an array using the FLOAT command. the
>> likes of FLTARR just keeps saying that I have more than 8 dimensions
>> and hence it won't work.
>
>> Any ideas guys?
>
>> Thanks
>

>> Will- Hide quoted text -
>
> - Show quoted text -

Sorry for impreciseness. I don't seem to be on the ball at all today.
Yeah that makes sense, and to answer your questions it is multiple
images I am loading into two seperate float arrays thats 122 images
for each array, so a grand total of 244 images are being loaded. The
idea being to compare the mass amount of images with each other and
moving the array around both in the x and y direction until the
closest match is found.

I hope that clears it up for you, as I say I am not really on the ball
today.

Thanks
Will

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by Will on Mon, 08 Mar 2010 12:44:20 GMT
View Forum Message <> Reply to Message

On Mar 8, 12:31 pm, Spon <christoph.b...@gmail.com> wrote:
> On Mar 8, 11:43 am, Will <theloneguitar...@hotmail.co.uk> wrote:
>
>
>
>
>
>
>> Hi
>
>> sorry I don't think the heading was very clear there.
>
>> I have loaded a group of arrays (jpeg files) into a seperate float
>> array, and again with another group of files I have done the same
>> thing. I am now trying to have the two arrays perform a subtraction
>> calculation with one another and what I want to do is have the arrays
>> do this with each other until they have did the calculation with every
>> element. i.e
>
>> it does the first calculation, then offsets the elements by one to the
>> right and does the next calculation then repeats this until it has
>> done every element in the array.
>
>> I was curious as to how to do this successflly I am currently trying
>> to use a FOR loop after my Repeat loop failed. I have the maths I need
>> here but I don't know how to access the subscripts of my array. I have

>> inputted all the files into an array using the FLOAT command. the
>> likes of FLTARR just keeps saying that I have more than 8 dimensions
>> and hence it won't work.
>
>> Any ideas guys?
>
>> Thanks
>
>> Will
>
> Hi Will,
>
> do you mean like this?:
>
> arr1 = [3, 4, 2, 8]
> arr2 = [1, 0, 4, 8]
> diff = arr1 - arr2
>
> print,diff
> 2 4 -2 0
>
> 'diff' will be a 4-element, one dimensional array in this case, but
> can just as easily be multidimensional.
> So long as the number of elements in both arrays are the same, simply
> using the '-' operator will do.
>
> Regards,
> Chris- Hide quoted text -
>
> - Show quoted text -

No i'm afraid not Chris, What I am doing is comparing two large amounts of images with each other but the images are in a specific order and can't be changed. These images then need to find the best fit location. So i am using a float array fo i can print out a single figure as the images try to find the best fit location but i need to access my subscripts and find the offset. so one set of these bundles will move over the other bundle to the right for instance doing a calculation to determine how similar they are (simple subtraction) and then move down a row and do the same thing. then with all these single point numbers i can make a plot and where the line is closest to 0 is where my images were best matched.

hope that jargon made sense

Will

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by jeanh on Mon, 08 Mar 2010 16:52:48 GMT
View Forum Message <> Reply to Message

> Sorry for impreciseness. I don't seem to be on the ball at all today.
> Yeah that makes sense, and to answer your questions it is multiple
> images I am loading into two seperate float arrays thats 122 images
> for each array, so a grand total of 244 images are being loaded. The
> idea being to compare the mass amount of images with each other and
> moving the array around both in the x and y direction until the
> closest match is found.
>
> I hope that clears it up for you, as I say I am not really on the ball
> today.
>
> Thanks
> Will

Hi Will,

ok, I get a better idea... do you want to move all your images at the same time and do the comparison, or one by one? (i.e., do you want to have a shift of let's say 1;5 for the 1st image, and 85,20 for the 2nd image, or do you want to move all your images by 2;5?)

Anyways, "shift" is your friend here. Be careful on the edge of the images... you might want to remove the edges, as values are wrapped around.

Now, depending on the content of your images, you can do things differently... like identifying a region of interest (function region), then identify the point of gravity and shift your images accordingly... but again, it all depends on the content of the images!

Jean

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by Juggernaut on Mon, 08 Mar 2010 16:56:53 GMT
View Forum Message <> Reply to Message

On Mar 8, 11:52 am, jeanh
<jghasb...@DELETETHIS.environmentalmodelers.ANDTHIS.com> wrote:
>> Sorry for impreciseness. I don't seem to be on the ball at all today.
>> Yeah that makes sense, and to answer your questions it is multiple
>> images I am loading into two seperate float arrays thats 122 images
>> for each array, so a grand total of 244 images are being loaded. The
>> idea being to compare the mass amount of images with each other and
>> moving the array around both in the x and y direction until the

>> closest match is found.
>
>> I hope that clears it up for you, as I say I am not really on the ball
>> today.
>
>> Thanks
>> Will
>
> Hi Will,
>
> ok, I get a better idea... do you want to move all your images at the
> same time and do the comparison, or one by one? (i.e., do you want to
> have a shift of let's say 1;5 for the 1st image, and 85,20 for the 2nd
> image, or do you want to move all your images by 2;5?)
>
> Anyways, "shift" is your friend here. Be careful on the edge of the
> images... you might want to remove the edges, as values are wrapped around.
>
> Now, depending on the content of your images, you can do things
> differently... like identifying a region of interest (function region),
> then identify the point of gravity and shift your images accordingly...
> but again, it all depends on the content of the images!
>
> Jean

Sounds like image registration is what you're really looking for...but
I could be wrong. There are a number of image registration algorithms
out there that work a bit more sophisticatedly to make things a bit
easier for you. Look up IDL Fourier Image Registration...a quick
algorithm for image registration. Using areas of interest instead of
the whole image will generally speed things up and give you the
required shifts using the Fourier method unless there are huge shifts
in the x and y directions. Hopefully I'm not too far off base for
you.
http://www.utsa.edu/lrsg/Teaching/EES5053-06/project/Cynthia .pdf

---

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by Will on Tue, 09 Mar 2010 14:19:31 GMT
View Forum Message <> Reply to Message

On 8 Mar, 16:52, jeanh
<jghasb...@DELETETHIS.environmentalmodelers.ANDTHIS.com> wrote:
>> Sorry for impreciseness. I don't seem to be on the ball at all today.
>> Yeah that makes sense, and to answer your questions it is multiple
>> images I am loading into two seperate float arrays thats 122 images
>> for each array, so a grand total of 244 images are being loaded. The
>> idea being to compare the mass amount of images with each other and

>> moving the array around both in the x and y direction until the
>> closest match is found.
>
>> I hope that clears it up for you, as I say I am not really on the ball
>> today.
>
>> Thanks
>> Will
>
> Hi Will,
>
> ok, I get a better idea... do you want to move all your images at the
> same time and do the comparison, or one by one? (i.e., do you want to
> have a shift of let's say 1;5 for the 1st image, and 85,20 for the 2nd
> image, or do you want to move all your images by 2;5?)
>
> Anyways, "shift" is your friend here. Be careful on the edge of the
> images... you might want to remove the edges, as values are wrapped around.
>
> Now, depending on the content of your images, you can do things
> differently... like identifying a region of interest (function region),
> then identify the point of gravity and shift your images accordingly...
> but again, it all depends on the content of the images!
>
> Jean

Hi Jean

I am just looking to shift all of the images on the first array and
keep all of the images on the second array the same. (Second array
does not shift at all).

I thought along those lines too but i'm not sure if i'm getting my
head around "SHIFT" properly. I have the program shifting the the
images as described but I get the sneaking impression it is doing the
wrong calculation or more likely the calculation isn't changing much.
Anyway to clarify, I would be correct in assuming that the images
loaded into the 1st array are sequential, they go from one through to
122 and that when I tell the array to shift by say 1, all of the
images move to the right by one image? or by one element of the images
array and not the array we have just created. If I am right about my
thinking and we are both agreed on "shift" then the program is doing
what it is meant to and outputting the correct answers which means
when I make a graph of the output I can tell where the best fit for
the images is and know the offset. I think :S.

Thank you for our help

Will

Subject: Re: IDL calculating elements in arrays plus there offsets
Posted by Will on Tue, 09 Mar 2010 14:21:15 GMT
View Forum Message <> Reply to Message

On 8 Mar, 16:56, Bennett <juggernau...@gmail.com> wrote:
> On Mar 8, 11:52 am, jeanh
>
>
>
>
>
> <jghasb...@DELETETHIS.environmentalmodelers.ANDTHIS.com> wrote:
>>> Sorry for impreciseness. I don't seem to be on the ball at all today.
>>> Yeah that makes sense, and to answer your questions it is multiple
>>> images I am loading into two seperate float arrays thats 122 images
>>> for each array, so a grand total of 244 images are being loaded. The
>>> idea being to compare the mass amount of images with each other and
>>> moving the array around both in the x and y direction until the
>>> closest match is found.
>
>>> I hope that clears it up for you, as I say I am not really on the ball
>>> today.
>
>>> Thanks
>>> Will
>
>> Hi Will,
>
>> ok, I get a better idea... do you want to move all your images at the
>> same time and do the comparison, or one by one? (i.e., do you want to
>> have a shift of let's say 1;5 for the 1st image, and 85,20 for the 2nd
>> image, or do you want to move all your images by 2;5?)
>
>> Anyways, "shift" is your friend here. Be careful on the edge of the
>> images... you might want to remove the edges, as values are wrapped around.
>
>> Now, depending on the content of your images, you can do things
>> differently... like identifying a region of interest (function region),
>> then identify the point of gravity and shift your images accordingly...
>> but again, it all depends on the content of the images!
>
>> Jean
>
> Sounds like image registration is what you're really looking for...but
> I could be wrong. There are a number of image registration algorithms

> out there that work a bit more sophisticatedly to make things a bit
> easier for you. Look up IDL Fourier Image Registration...a quick
> algorithm for image registration. Using areas of interest instead of
> the whole image will generally speed things up and give you the
> required shifts using the Fourier method unless there are huge shifts
> in the x and y directions. Hopefully I'm not too far off base for
> you.http://www.utsa.edu/lrsg/Teaching/EES5053-06/project/Cyn thia.pdf- Hide quoted text -
>
> - Show quoted text -

Hi Bennet

Thanks for the input I have got to admit I have not looked into image registration, and shall do so immediately starting with the link you sent me. Thanks again.

Will
