
Subject: Finding the mean of a set of images
Posted by [David Oesch](#) on Tue, 22 Oct 2002 08:45:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello outthere,

I know this topic was up before, but all I could find in the list was to go for CALL_EXTERNAL and use a FORTRAN etc. program. Here's my problem: Does anyone have an algorithm for finding the mean/standarddeviation etc at each pixel position for a set of equal size 2-D images? Currently the only way I have to do this is to extract all the values for a given pixel position into a 1-D array and find the mean/standarddeviation etc on that. Doing it pixel by pixel like this is inefficient in IDL so I am looking for an *array* based algorithm that would find all the mean/standarddeviation etc in parallel. Any progs so far in IDL for this problem?..or a decent fortran or C program?

Cheers...

Dave

--

David Oesch - Remote Sensing Research Group

Department of Geography

University of Bern Tel : +41 (0)31 631 8020
Hallerstr. 12 Fax : +49 (0)89 2443 43780
CH - 3012 Bern Mail: David.Oesch@giub.unibe.ch
Switzerland <http://www.giub.unibe.ch/remsen>

Remote Sensing is...
Staying as far away from the problem as possible.
- G. Archer, World Bank

Subject: Re: Finding the mean of a set of images
Posted by [R.Bauer](#) on Wed, 23 Oct 2002 16:54:55 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Oesch wrote:

>
>
> Jaco van Gorkom wrote:

```

>
>> "Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote in message
>> news:onznt6k38x.fsf@cow.physics.wisc.edu...
>>
>>> David Oesch <oesch@giub.unibe.ch> writes:
>>>
>>>> ...
>>>> Does anyone have an algorithm for finding the mean/standard deviation etc
>>>> at each pixel position for a set of equal size 2-D images? Currently the
>>>> only way I have to do this is to extract all the values for a given
>>>> pixel position into a 1-D array and find the mean/standard deviation etc
>>>> on that. Doing it pixel by pixel like this is inefficient in IDL so I am
>>>> looking for an *array* based algorithm that would find all
>>>> the mean/standard deviation etc in parallel. ...
>>>>
>>> Sure, if you stack your image into a 3D image cube, then you would
>>> have something like IMAGE = FLTARR(NX, NY, NIMAGES)
>>>
>>> Then the mean image is:
>>>
>>> mean = total(image,3)/nimages
>>>
>>> The standard deviation is:
>>>
>>> meancube = rebin(reform(mean,nx,ny,1),nx,ny,nimages)
>>> std = sqrt(total((image - meancube)^2,3)/(nimages-1))
>>>
>>> Now, what you meant by "etc" can get a little hairier. If you want to
>>> do median you are probably in trouble, but min and max are easy too:
>>>
>>> minimage = image(*,*,0)
>>> maximage = minimage
>>> for i = 1, nimages-1 do begin
>>>   minimage = minimage < image(*,*,i)
>>>   maximage = maximage > image(*,*,i)
>>> endfor
>>>
>>
>> I believe that IDL 5.5 offers the luxury of
>> maximage = MAX(image, MIN=minimage, DIMENSION=3)
>>
>> As for median, well, there was a thread on "Finding the median of a set
>> of images" back in '96. Should be valid still, I guess. Improvising with
>> transpose(), reform(), median(image, Nimages) and rebin() should be fun,
>> but not be very fast.
>>
>> Jaco
>>

```

>>
> Well it works fine in 5.5 , but this feature is not explained in the help
> of the idl 5.5 WIN/PC edition
>
> ... in 5.5
> the help is obviously still the help of 5.4...

Yes

and there was a big threat already about this

Ben and Andrew have developed their cool_help tool

which is a idl widget to search to the entire idl5.5 help.

[http://groups.google.de/groups?hl=de&lr=&ie=UTF-8&am
p;threadm=pan.2002.07.22.18.34.47.124252.19863%40as.arizona.
edu&rnum=9&prev=/groups%3Fq%3Dhelp%2B%252B2002%2Bgro
up:comp.lang.idl-pvwave%26hl%3Dde%26lr%3D%26ie%3DUTF-8%26sel
m%3Dpan.2002.07.22.18.34.47.124252.19863%2540as.arizona.edu%26rnum%3D9
www.tidewater.net/~pemaquid](http://groups.google.de/groups?hl=de&lr=&ie=UTF-8&am
p;threadm=pan.2002.07.22.18.34.47.124252.19863%40as.arizona.
edu&rnum=9&prev=/groups%3Fq%3Dhelp%2B%252B2002%2Bgro
up:comp.lang.idl-pvwave%26hl%3Dde%26lr%3D%26ie%3DUTF-8%26sel
m%3Dpan.2002.07.22.18.34.47.124252.19863%2540as.arizona.edu%26rnum%3D9
www.tidewater.net/~pemaquid)

All changes are described in the whats_new pdf file too.

Reimar

--

Forschungszentrum Juelich

email: R.Bauer@fz-juelich.de

<http://www.fz-juelich.de/icg/icg-i/>

=====

a IDL library at Forschungszentrum Juelich

http://www.fz-juelich.de/icg/icg-i/idl_icglib/idl_lib_intro.html

Subject: Re: Finding the mean of a set of images

Posted by [mfeldt](#) on Mon, 28 Oct 2002 09:06:48 GMT

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

Hi Dave,

it's not easy to find an entirely array based solution for this.
for me, i found that it helps a great deal in speed if i collapse the
cube to two dimensions and then have a loop run over only *one*
dimension (instead of two nested loops, which are really slow in
idl). the additional advantage is, that in this context you can do
all sorts of things to your cube besides mean and sd - like median,
certain percentiles, etc...

looks somewhat like this:

```
InArr = fltarr(sx, sy, n) ; this is the input array
OutArr = fltarr(sx*sy)
UseArr = reform(InArr, sx*sy, n)
for i=0l, sx*sy-1 do $
  OutArr[i]=median(InArr[i,*]) ; in this case, computes the median
                                ; can stuff other operations here
OutArr=reform(OutArr,sx,sy,/over)
```

have fun with it

mfeldt

--

Markus Feldt	Voice:	+49 6221 528 262
Max-Planck-Institut	Fax:	+49 6221 528 246
fÄr Astronomie		mailto:mfeldt@mpia.de
KÄnigstuhl 17		http://www.mpia.de/homes/feldt
D-69117 Heidelberg, Germany		Si, !asi es la vida!

David Oesch <oesch@giub.unibe.ch> wrote in message
news:<3DB51012.6070200@giub.unibe.ch>...

```
> Hello outthere,
>
> I know this topic was up before, but all I could find in the list was to
> go for CALL_EXTERNAL and use a FORTRAN etc. program. Here's my problem:
> Does anyone have an algorithm for finding the mean/standarddeviation etc
> at each pixel position for a set of equal size 2-D images? Currently the
> only way I have to do this is to extract all the values for a given
> pixel position into a 1-D array and find the mean/standarddeviation etc
> on that. Doing it pixel by pixel like this is inefficient in IDL so I am
> looking for an *array* based algorithm that would find all
> the mean/standarddeviation etc in parallel.
> Any progs so far in IDL for this problem?..or a decent fortran or C program?
>
> Cheers...
>
> Dave
>
> --
```

> _____
>
> David Oesch - Remote Sensing Research Group
>
> Department of Geography
>
> University of Bern Tel : +41 (0)31 631 8020
> Hallerstr. 12 Fax : +49 (0)89 2443 43780
> CH - 3012 Bern Mail: David.Oesch@giub.unibe.ch
> Switzerland http://www.giub.unibe.ch/remsen
>
> Remote Sensing is...
> Staying as far away from the problem as possible.
> - G. Archer, World Bank
> _____

Subject: Re: Finding the mean of a set of images
Posted by [David Oesch](#) on Mon, 28 Oct 2002 10:06:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi, mfedt

Thanks a lot..I think this is the easiest way...even if reform ain't that fast, but still it works... instead the FOR loop, WHERE is even more powerfull ...

Dave

mfeldt wrote:

> Hi Dave,
>
> it's not easy to find an entirely array based solution for this.
> for me, i found that it helps a great deal in speed if i collapse the
> cube to two dimensions and then have a loop run over only *one*
> dimension (instead of two nested loops, which are really slow in
> idl). the additional advantage is, that in this context you can do
> all sorts of things to your cube besides mean and sd - like median,
> certain percentiles, etc...
>
> looks somewhat like this:
>
>
> InArr = fltarr(sx, sy, n) ; this is the input array
> OutArr = fltarr(sx*sy)
> UseArr = reform(InArr, sx*sy, n)
> for i=0l, sx*sy-1 do \$

>> University of Bern Tel : +41 (0)31 631 8020
>> Hallerstr. 12 Fax : +49 (0)89 2443 43780
>> CH - 3012 Bern Mail: David.Oesch@giub.unibe.ch
>> Switzerland http://www.giub.unibe.ch/remsen
>>
>> Remote Sensing is...
>> Staying as far away from the problem as possible.
>> - G. Archer, World Bank
>>

>>

--

David Oesch - Remote Sensing Research Group

Department of Geography

University of Bern Tel : +41 (0)31 631 8020
Hallerstr. 12 Fax : +49 (0)89 2443 43780
CH - 3012 Bern Mail: David.Oesch@giub.unibe.ch
Switzerland http://www.giub.unibe.ch/remsen

Remote Sensing is...
Staying as far away from the problem as possible.
- G. Archer, World Bank