Subject: Re: image subtraction
Posted by Dick Jackson on Thu, 14 Mar 2002 02:48:54 GMT
View Forum Message <> Reply to Message

"Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote in message
news:onbsdsxalv.fsf@cow.physics.wisc.edu...
>
> aburden@mpl.ucsd.edu (Art) writes:
>> Hi,
>>
>> Can anyone come up with an efficient way to subtract a single image
>> from an array of images without using a loop, as in
>>
>> imgArr = imgArr - img1
>>
>> where,
>>
>> imgArr = lonarr(12,512,512)
>>
>> and,
>>
>> img1 = lonarr(512,512)
>>
>> as opposed to
>>
>> For ii=0,11 do imgArr[ii,*,*] = imgArr[ii,*,*] - img1
>
> (a) You can do this:
> imgarr = imgarr - rebin(reform(img1,1,512,512),12,512,512)
>
> This makes IMG1 into a 1x512x512 array, and then uses REBIN to
> expand it to the desired size.
>
> (b) I don't think a FOR loop will be that much slower than the REBIN
> approach, at least for 512x512 arrays.
>
> Good luck,
> Craig

I second Craig's suggestion, and add a small addendum:

The assignment in the FOR loop can be sped up a bit by writing it as:

FOR ii=0,11 DO imgArr[ii,0,0] = imgArr[ii,*,*] - img1
                ^ ^
The values will be laid in as expected, but this dropped my timing from
1.402 seconds to 1.232 seconds.

Now, having said that, the REBIN method (useful if you have enough memory to make the second big array) comes in at 0.210 seconds!

Cheers,
--
-Dick

Dick Jackson / dick@d-jackson.com
D-Jackson Software Consulting / http://www.d-jackson.com
Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392