Subject: Re: array convol optimization slow now Posted by David Fanning on Mon, 13 Jul 2009 22:57:17 GMT

View Forum Message <> Reply to Message

## newerjazz writes:

- > I have an array of [2,100,000,000] corresponding to x, y and
- > 100,000,000 particles. I need to render these particles as gasussian
- > spots at x, y locations

>

- > Currently, I run a for loop through each particle and render each
- > particle in this for loop; it takes forever.

Not to mention it probably takes a display screen the size of a city block to render the mess in the kind of resolution that allows you to actually make something of it.

Have you tried plotting a few hundred (maybe even a few thousand) random points to see if you can draw the same conclusions you were drawing by plotting all the points?

Cheers.

David

--

David Fanning, Ph.D.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: array convol optimization slow now Posted by newerjazz on Mon, 13 Jul 2009 23:10:15 GMT View Forum Message <> Reply to Message

Thanks David,

each particle is a single molecule of protein; we need to visualize all the proteins to get meaningful interpretation.

i can look at 1/10 of the particles by reducing the field of view but it's still slow.

do you have any suggestions of getting the x,y locations on the image w/ an array operation instead of a for loop?

## Thanks!

## newerjazz

```
On Jul 13, 3:57 pm, David Fanning <n...@dfanning.com> wrote:
> newerjazz writes:
>> I have an array of [2,100,000,000] corresponding to x, y and
>> 100,000,000 particles. I need to render these particles as gasussian
>> spots at x, y locations
>
>> Currently, I run a for loop through each particle and render each
>> particle in this for loop; it takes forever.
>
> Not to mention it probably takes a display screen
> the size of a city block to render the mess in
> the kind of resolution that allows you to actually
> make something of it.
> Have you tried plotting a few hundred (maybe even
> a few thousand) random points to see if you can draw
> the same conclusions you were drawing by plotting all
> the points?
> Cheers.
> David
> David Fanning, Ph.D.
> Coyote's Guide to IDL Programming (www.dfanning.com)
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
```

Subject: Re: array convol optimization slow now Posted by David Fanning on Mon, 13 Jul 2009 23:26:02 GMT View Forum Message <> Reply to Message

## newerjazz writes:

- > each particle is a single molecule of protein; we need to visualize
- > all the proteins to get meaningful interpretation.

How does that work? If I put 100 million dots on my screen, things get a little blurry. (Of course, my eyes aren't what they used to be, admittedly.)

- > do you have any suggestions of getting the x,y locations on the image
- > w/ an array operation instead of a for loop?

No, I can't think of any way to do this an an array. :-(
Cheers,

David
-David Fanning, Ph.D.
Coyote's Guide to IDL Programming (www.dfanning.com)
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: array convol optimization slow now Posted by Chris[6] on Mon, 13 Jul 2009 23:53:53 GMT View Forum Message <> Reply to Message

On Jul 13, 1:26 pm, David Fanning <n...@dfanning.com> wrote: > newerjazz writes: >> each particle is a single molecule of protein; we need to visualize >> all the proteins to get meaningful interpretation. > > How does that work? If I put 100 million dots on my screen, things > get a little blurry. (Of course, my eyes aren't what they used to > be, admittedly.) >> do you have any suggestions of getting the x,y locations on the image >> w/ an array operation instead of a for loop? > No, I can't think of any way to do this an an array. :-( > Cheers, > David > David Fanning, Ph.D. > Coyote's Guide to IDL Programming (www.dfanning.com) > Sepore ma de ni thui. ("Perhaps thou speakest truth.")

I've written a routine called densitymap which makes smoothed surface density maps based on the position of 2d sources. You might want to check it out (it has been moderately debugged)

http://www.ifa.hawaii.edu/users/beaumont/code/

It uses a different smoothing scheme (it uses the distance to each pixel's n-th nearest neighbor to estimate the surface density).

If you use it, you should probably download the IDL astronomy user's

library, as well as http://www.ifa.hawaii.edu/~beaumont/code/beaumont\_library.ta r to avoid dependency issues.

Otherwise, you could use hist\_2d to pixellate the locations of each point, and then convolve the histogram with a gaussian

Chris

Subject: Re: array convol optimization slow now Posted by newerjazz on Tue, 14 Jul 2009 15:31:15 GMT View Forum Message <> Reply to Message

Thanks Chris,

I will try hist\_2d to see if it helps.

Dave, we know the x,y position down to subpixel resolution; so instead of say plotting particles on a 512x512 grid; we plot it on 5120x5120 grid (if our resolution is 1/10 of a pixel). In practice our resolution is down to 1/100th of a pixel. Depending on how many particles are packed into a pixel or a voxel since I actually have z resolution too; the image can look either sharp or blur.

cheers, newerjazz

```
On Jul 13, 4:53 pm, Chris <beaum...@ifa.hawaii.edu> wrote:

> On Jul 13, 1:26 pm, David Fanning <n...@dfanning.com> wrote:

> 
> 
> 
> newerjazz writes:

>>> each particle is a single molecule of protein; we need to visualize

>>> all the proteins to get meaningful interpretation.

> 
> How does that work? If I put 100 million dots on my screen, things

>> get a little blurry. (Of course, my eyes aren't what they used to

>> be, admittedly.)

>>> do you have any suggestions of getting the x,y locations on the image

>>> w/ an array operation instead of a for loop?

> 
>> No, I can't think of any way to do this an an array. :-(

> Cheers,
```

```
>> David
>> --
>> David Fanning, Ph.D.
>> Coyote's Guide to IDL Programming (www.dfanning.com)
>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
> I've written a routine called densitymap which makes smoothed surface
> density maps based on the position of 2d sources. You might want to
> check it out (it has been moderately debugged)
>
> http://www.ifa.hawaii.edu/users/beaumont/code/
> It uses a different smoothing scheme (it uses the distance to each
 pixel's n-th nearest neighbor to estimate the surface density).
>
> If you use it, you should probably download the IDL astronomy user's
> library, as well_ashttp://www.ifa.hawaii.edu/~beaumont/code/beaumont_library.tar
> to avoid dependency issues.
>
> Otherwise, you could use hist_2d to pixellate the locations of each
 point, and then convolve the histogram with a gaussian
> Chris
```