Subject: write more efficient for loop
Posted by g.nacarts on Fri, 07 Aug 2015 13:25:06 GMT
View Forum Message <> Reply to Message

Hi

I used the Profiler, /SYSTEM & Profiler to identify which part of the code takes long time. I am using nested for loop (takes about 1950 sec).

```
N = 215L

Image_2 = make_array(N,N,/double)

for i=0L, N-1 do begin
  for j=0L, N-1 do begin
 x = round(i + A[i, j])
 y = round(j + B[i, j])
 if (x ge 0) && (y ge 0) then begin
      Image_2[i,j] = Image_1[x, y]
 endif
 endfor
endfor
```

I was wondering if there is a more efficient way to write the above. I don't have a lot of experience in programming.

Subject: Re: write more efficient for loop
Posted by Michael Galloy on Fri, 07 Aug 2015 13:50:07 GMT
View Forum Message <> Reply to Message

On 8/7/15 7:25 AM, g.nacarts@gmail.com wrote:

```
>  Hi
>
>  I used the Profiler, /SYSTEM & Profiler to identify which part of
>  the  code takes long time. I am using nested for loop (takes about 1950 sec).
>
>  N = 215L
>
>  Image_2 = make_array(N,N,/double)
>
>  for i=0L, N-1 do begin
>     for j=0L, N-1 do begin
>   x = round(i + A[i, j])
>   y = round(j + B[i, j])
>   if (x ge 0) && (y ge 0) then begin
>         Image_2[i,j] = Image_1[x, y]
>   endif
```

> endfor
> endfor
>
> I was wondering if there is a more efficient way to write the above.
> I don't have a lot of experience in programming.

Is image_1 n by n also? Here's some totally untested code, but hopefully
it provides some inspiration:

```
x = round(rebin(reform(findgen(n), 1), n, n) + a)
y = round(rebin(reform(1, findgen(n)), n, n) + b)
ind = where(x ge 0 and y ge 0, count)
if (count gt 0) then begin
  image_2[ind] = image_1[x[ind], y[ind]]
endif
```

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Subject: Re: write more efficient for loop
Posted by g.nacarts on Fri, 07 Aug 2015 13:55:05 GMT
View Forum Message <> Reply to Message

Yes Image_1 is also N by N.

Subject: Re: write more efficient for loop
Posted by g.nacarts on Fri, 07 Aug 2015 19:39:15 GMT
View Forum Message <> Reply to Message

I tried to understand what you suggested but it doesn't make a lot of sense to me.

I found in the link below that by exchanging the two FOR loops the computation time can be reduced
 http://www.exelisvis.com/Support/HelpArticlesDetail/TabId/219/ArtMID/900/ArticleID/1799/1799.aspx

I tried it but doesn't reduce at all. Maybe I did a mistake or I didn't understand well?

```
N = 215L

Image_2 = make_array(N,N,/double)
```

```
for j=0L, N-1 do begin
  for i=0L, N-1 do begin
        x = round(i + A[i, j])
        y = round(j + B[i, j])
        if (x ge 0) && (y ge 0) then begin
            Image_2[i,j] = Image_1[x, y]
        endif
 endfor
endfor
```

Subject: Re: write more efficient for loop
Posted by Michael Galloy on Mon, 10 Aug 2015 01:26:48 GMT
View Forum Message <> Reply to Message

On 8/7/15 1:39 PM, g.nacarts@gmail.com wrote:
> I tried to understand what you suggested but it doesn't make a lot of sense to me.
>
> I found in the link below that by exchanging the two FOR loops the computation time can be reduced
> http://www.exelisvis.com/Support/HelpArticlesDetail/TabId/219/ArtMID/900/ArticleID/1799/1799.aspx
>
> I tried it but doesn't reduce at all. Maybe I did a mistake or I didn't understand well?
>
>
> N = 215L
>
> Image_2 = make_array(N,N,/double)
>
> for j=0L, N-1 do begin
>   for i=0L, N-1 do begin
>   x = round(i + A[i, j])
>   y = round(j + B[i, j])
>   if (x ge 0) && (y ge 0) then begin
>   Image_2[i,j] = Image_1[x, y]
>   endif
>   endfor
> endfor
>

I had a couple typos in the x =... and y =... lines, check out the process for n=5:

```
IDL> n = 5
IDL> a = randomu(seed, n, n)
IDL> b = randomu(seed, n, n)
```

```
IDL> print, a
     0.671444     0.339387    0.0291560     0.626489     0.327068
     0.880495     0.773669     0.845125     0.133505     0.611774
     0.133046     0.206592     0.892692     0.924602     0.300541
     0.970698     0.572081     0.609453     0.881446     0.627840
     0.574902    0.0269283     0.326179     0.282648     0.197054
IDL> print, b
     0.179979     0.666919     0.447893     0.598895     0.747911
     0.665888     0.923736     0.571645     0.833802     0.205634
     0.539740     0.445450     0.643078    0.0266434     0.399671
     0.978612     0.377670     0.645917     0.332302     0.734567
     0.794310     0.146270     0.731570     0.971320     0.359303
IDL> x = round(rebin(reform(findgen(n), n, 1), n, n) + a)
IDL> y = round(rebin(reform(findgen(n), 1, n), n, n) + b)
IDL> print, x
           1           1           2           4           4
           1           2           3           3           5
           0           1           3           4           4
           1           2           3           4           5
           1           1           2           3           4
IDL> print, y
           0           1           0           1           1
           2           2           2           2           1
           3           2           3           2           2
           4           3           4           3           4
           5           4           5           5           4
IDL> ind = where(x ge 0 and y ge 0, count)
IDL> print, count
          25
IDL> image_1 = randomu(seed, n, n)
IDL> image_2 = fltarr(n, n)
IDL> image_2[ind] = image_1[x[ind], y[ind]]
IDL> print, image_1
    0.0991742     0.568827     0.928698     0.431867     0.582845
     0.570439     0.752998     0.347453     0.864314     0.397977
    0.0470480     0.847442     0.691785     0.343657    0.0733360
     0.418571     0.152661     0.580832     0.130468     0.283944
   0.00865856     0.547303     0.157060     0.316100     0.770079
IDL> print, image_2
     0.568827     0.752998     0.928698     0.397977     0.397977
     0.847442     0.691785     0.343657     0.343657     0.397977
     0.418571     0.847442     0.130468    0.0733360    0.0733360
     0.547303     0.580832     0.316100     0.283944     0.770079
     0.547303     0.547303     0.157060     0.316100     0.770079
IDL> print, ind
           0           1           2           3           4
5           6           7
           8           9          10          11          12
```

13 14 15
16 17 18 19 20
21 22 23
24

Mike
--
Michael Galloy
www.michaelgalloy.com
Modern IDL: A Guide to IDL Programming (http://modernidl.idldev.com)

Subject: Re: write more efficient for loop
Posted by g.nacarts on Mon, 10 Aug 2015 15:38:54 GMT
View Forum Message <> Reply to Message

That's great. Thank you very much!
