
Subject: Avoiding FOR loops (version googleplex.infinity)

Posted by [Gaurav](#) on Mon, 07 Apr 2008 12:42:57 GMT

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

Dear all,

I know IDL provides us with neat tricks to avoid the processor monster called FOR loop but I always tend to become incapable using the magic tricks viz WHERE when I need them the most. Will someone please show me the way here.

What I have is a 2D Byte array. For each element I need to find the number of elements in a surrounding 5 X 5 window that is equal to the element under scrutiny. It is easy to achieve this using FOR loops and EXTRAC & WHERE functions, but it obviously becomes very slow for large arrays.

How do I avoid the FOR loop in this case using wildcards and the ":" operator?

Like always, I hope I do not get to gnaw at my fingernails for long.

Cheers
Gaurav

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [Gaurav](#) on Wed, 09 Apr 2008 12:33:46 GMT

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

Dear Nathan,

And how do you propose I take care of the pixels where the 'chunks' will join other 'chunks'?

I have thought of a very neat algorithm today, but will be able to implement it only later today. I promise to come up with some positive news tomorrow.

Cheers,
Gaurav

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [MichaelT](#) on Wed, 09 Apr 2008 12:56:32 GMT

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

On Apr 9, 2:33 pm, Gaurav <selfishgau...@gmail.com> wrote:

> Dear Nathan,
>
> And how do you propose I take care of the pixels where the 'chunks'
> will join other 'chunks'?
>
> I have thought of a very neat algorithm today, but will be able to
> implement it only later today. I promise to come up with some positive
> news tomorrow.
>
> Cheers,
> Gaurav

Hi Gaurav,

you could simply use some overlap.

Generate some vectors which contain the start and end positions of each chunk.

Something like:

```
pxs = [0, p1-sr, p2-sr, p3-sr, ...]
```

```
pxe = [p1, p2, p3, ...]
```

and the same for y-coordinates

So you simply need a loop now and take the respective chunks from your original array:

```
a = b[pxs[i]: pxe[i], pys[i]: pye[i]]
```

Then you will get the correct values for each of your chunks. So in your case it seems to be impossible without loops...

What about Jim's method? What does that one look like? There could be better solutions than mine.

Michael

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [Tom McGlynn](#) on Thu, 10 Apr 2008 18:49:22 GMT

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

On Apr 10, 2:13 am, Gaurav <selfishgau...@gmail.com> wrote:

... Now it is up to you guys to decide if it

> is exactly the same as Tom's or there is some difference and as to

> which would be faster.

Nah... You're the one with the problem. Why should we do your work for you! (: Seriously, it's up to you to decide which you prefer for whatever reason. The code I wrote doesn't create as many temporary arrays, so I might expect it to be a bit faster, but if the number of

matches is small then yours accesses your output array less. Perhaps that balances things. Measure it and see. I do believe that the two give different answers on the edges of the image, but which (if either) is correct depends upon details of the specification of the problem.

In any case it's a little ironic that your code illustrates one of the common IDL bugs that I mentioned in my post on the concurrent thread on that topic in this group. Can you find it?

Good luck,
Tom McGlynn

```
> ...Here follows my code:
>
> ##### Code begin #####
> dims = size(img,/dimensions) ; getting the dimensions of input image
> output_img = bytarr(dims(0), dims(1))
>
> kernelSize = 5 ;define the kernel size (here, 5)
> padSize = FLOOR(kernelSize /2.0) ;calculate the padding to generate
> around the original image
> paddedImg = bytarr(dims(0)+2*padSize, dims(1)+2*padSize) ;initialize
> the padded image
> paddedImg(padSize, padSize) = img ;define the padded image
> templmg = bytarr(dims(0)+2*padSize, dims(1)+2*padSize) ;will contain
> the padded, final output
>
> for x = -padSize, padSize do begin
>     for y = -padSize, padSize do begin
>         if x eq 0 and y eq 0 then continue ;we do not want to compare with
> the element itself
>         ;the x, y loops shift the whole image array around the kernel and
> compares the shifted image with the original
>         indices = WHERE(paddedImg eq SHIFT(paddedImg, x, y)) ;
>         templmg(indices) = finallmg(indices) + 1 ;increment '1' wherever an
> equality is found
>         endfor
>     endfor
> output_img = EXTRAC(templmg, padSize, padSize, dims(0),
> dims(1)) ;extract the output image from the padded output image.
> ##### end of code/algorithm #####
>
> In my case it works wonderfully well and speeds things up by a factor
> of a few hundred times. Any further optimization would be highly
> appreciated.
>
> I guess, I ought to go for a manicure for my gnawed nails now.
```

>
> Cheers,
> Gaurav

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [Gaurav](#) on Fri, 11 Apr 2008 06:15:39 GMT

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

Dear Tom,

The trouble is that coming into this group, I always feel I am a novice with IDL programming and algorithm designing. Hence, my reluctance in asserting my views.

For example, my perusal of IDL help documents has led me to the belief that creating temporary arrays help hasten things up while you mention something to the contrary. I shall definitely look this up and would appreciate pointers in the right direction.

As far as your code generating a result different from mine at the edges is concerned, that-of course-is true. In that respect, my code produces the same effect as the normal FOR loop, and this has been verified.

And as for the bug in my code, I definitely stand corrected for my code would fail if all the array elements in the kernel are different. Thanks indeed for pointing it out to me.

Thanks indeed,
Gaurav

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [weitkamp](#) on Fri, 11 Apr 2008 07:52:05 GMT

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

Gaurav,

I suggest a solution that is essentially identical to yours, with the sole difference that I think the use of WHERE is superfluous because it is much easier to use only its argument. (Using WHERE in cases where its argument alone is much easier to handle is actually one of the most frequent programming flaws that I have seen in this group, and I was tempted to submit to David Fanning's recent survey.) Below is the code I propose.

Cheers,

Timm

count = 0B * image ; This will contain the result

```
FOR ix = -width/2, width/2 DO BEGIN
  FOR iy = -height/2, height/2 DO BEGIN
    IF (ix NE 0) OR (iy NE 0) THEN BEGIN
      shiftedImage = SHIFT(image, ix, iy)
      count = count + (shiftedImage EQ image)
    ENDIF
  ENDFOR
ENDFOR

END
```

Gaurav a écrit :

> Dear Tom,
> The trouble is that coming into this group, I always feel I am a
> novice with IDL programming and algorithm designing. Hence, my
> reluctance in asserting my views.
>
> For example, my perusal of IDL help documents has led me to the belief
> that creating temporary arrays help hasten things up while you mention
> something to the contrary. I shall definitely look this up and would
> appreciate pointers in the right direction.
>
> As far as your code generating a result different from mine at the
> edges is concerned, that-of course-is true. In that respect, my code
> produces the same effect as the normal FOR loop, and this has been
> verified.
>
> And as for the bug in my code, I definitely stand corrected for my
> code would fail if all the array elements in the kernel are different.
> Thanks indeed for pointing it out to me.
>
> Thanks indeed,
> Gaurav

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [Gaurav](#) on Fri, 11 Apr 2008 12:41:45 GMT

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

> I suggest a solution that is essentially identical to yours, with the
> sole difference that I think the use of WHERE is superfluous because
> it is much easier to use only its argument. (Using WHERE in cases
> where its argument alone is much easier to handle is actually one of
> the most frequent programming flaws that I have seen in this group,

> and I was tempted to submit to David Fanning's recent survey.) Below
> is the code I propose.
>
> Cheers,
> Timm

Indeed the WHERE is superfluous! This one reduced the run time by another half. Thanks indeed for your input.

Wow, I have come a long way from thinking that the BoxCar approach was the only one that would be applicable in my case to this, five line wondercode that achieves exactly the same thing in a miniscule fraction of the time taken. This is why I am in love with this group.

Thanks indeed all of you...

Cheers
Gaurav

Subject: Re: Avoiding FOR loops (version googleplex.infinity)
Posted by [David Fanning](#) on Fri, 11 Apr 2008 12:56:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Gaurav writes:

> The trouble is that coming into this group, I always feel I am a
> novice with IDL programming and algorithm designing. Hence, my
> reluctance in asserting my views.

Get over it.

My experience has been that IDL knowledge increases in direct proportion to the user's willingness to hit the Send button. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

On Apr 11, 2:15 am, Gaurav <selfishgau...@gmail.com> wrote:

> Dear Tom,
> The trouble is that coming into this group, I always feel I am a
> novice with IDL programming and algorithm designing. Hence, my
> reluctance in asserting my views.
>
> For example, my perusal of IDL help documents has led me to the belief
> that creating temporary arrays help hasten things up while you mention
> something to the contrary. I shall definitely look this up and would
> appreciate pointers in the right direction.
>
> As far as your code generating a result different from mine at the
> edges is concerned, that-of course-is true. In that respect, my code
> produces the same effect as the normal FOR loop, and this has been
> verified.
>
> And as for the bug in my code, I definitely stand corrected for my
> code would fail if all the array elements in the kernel are different.
> Thanks indeed for pointing it out to me.
>
> Thanks indeed,
> Gaurav

Dear Guarav,

I learn something new in almost every discussion I participate on in this group -- and that's ignoring all of the stuff on IDL objects which I promptly forget. Don't assume that just because we're willing to pontificate we really know what we're talking about. With that caveat...

With regard to the use of temporaries, the big savings -- the factors of 100 -- come when you can replace some inner loop with an array expression.

E.g., we don't want to have 100,000,000 iterations over all of the pixels of your 10,000x10,000 images. However computing arrays has some cost in both CPU and memory footprint: so you don't want to create them willy nilly. If could just reuse existing arrays, then maybe that will be your best shot.

One thing that I didn't consider a couple of messages up is that a simple array expression

$x = y + z$

may be somewhat faster than a bounded array expression

$$x[1:10] = y[0:9] + z[21:29]$$

so it may be -- I don't know -- that creating temporaries so that you can use a simple expression will be a net benefit. It's the cost of building the new arrays versus the cost of the more complex computation. But I think differences are going to be quite modest -- factors of 2 at best I'd think. I suspect the only way to tell if the benefit of being able to simpler array operations outweighs the cost of building intermediate temporary arrays is to try it out with your hardware and your problem.

Regards,
Tom McGlynn

Subject: Re: Avoiding FOR loops (version googleplex.infinity)
Posted by [R.G. Stockwell](#) on Fri, 11 Apr 2008 16:15:51 GMT
[View Forum Message](#) <> [Reply to Message](#)

"David Fanning" <news@dfanning.com> wrote in message
news:MPG.22690dc6721c55ca98a328@news.frii.com...

> Gaurav writes:

>

>> The trouble is that coming into this group, I always feel I am a
>> novice with IDL programming and algorithm designing. Hence, my
>> reluctance in asserting my views.

>

> Get over it.

>

> My experience has been that IDL knowledge increases
> in direct proportion to the user's willingness to
> hit the Send button. :-)

True, in fact hitting Send is almost always followed
by an immediate epiphany on why my post is dumb.

:)

cheers,
bob

Subject: Re: Avoiding FOR loops (version googleplex.infinity)
Posted by [David Fanning](#) on Fri, 11 Apr 2008 16:44:55 GMT

R.G. Stockwell writes:

> True, in fact hitting Send is almost always followed
> by an immediate epiphany on why my post is dumb.
>
> :)

Well, of course. But who are we to question the laws of the Universe. :-)

Cheers,

David

P.S. I admit it is a strange way to learn, and throws immediately into question why we spent so many fruitless years in college. But it is amazingly effective.

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Seppure ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [Greg Hennessy](#) on Fri, 11 Apr 2008 17:32:39 GMT

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

On 2008-04-11, David Fanning <news@dfanning.com> wrote:

> P.S. I admit it is a strange way to learn, and throws
> immediately into question why we spent so many fruitless
> years in college.

I thought we went to college to drink beer.

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [David Fanning](#) on Fri, 11 Apr 2008 17:38:29 GMT

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

Greg Hennessy writes:

> I thought we went to college to drink beer.

I had something a little curvier in mind. But, yes,

not all of us were there for the right reasons. :-(

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Avoiding FOR loops (version googleplex.infinity)

Posted by [Gaurav](#) on Tue, 15 Apr 2008 05:46:07 GMT

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

> I had something a little curvier in mind. But, yes,

> not all of us were there for the right reasons. :-(

Yes, As David says it: "Sepore ma de ni thui."
