

---

Subject: Re: Matlab Translations -- Was: filtering images in the fourier domain  
Posted by [Matt Feinstein](#) on Tue, 18 Nov 2003 13:11:58 GMT

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

---

Since I use both Matlab and IDL (although, usually, not both at the same time) I can make some comments here--

On Mon, 17 Nov 2003 17:47:12 -0700, David Fanning  
<david@dfanning.com> wrote:

> The most obvious difference is that Matlab array indexing  
> starts at 1. This causes a lot of errors in my loops. And  
> I see LOTS of loops in the Matlab code. I usually  
> start by doing a direct (you might say "loopy") translation  
> to be sure I'm getting the code correct. Then I usually  
> say "Wait a minute. Can't those 10 lines be replaced with  
> this one?". The answer is inevitably "yes".

The start-at-1 rather than start-at-0 thing: has been the subject of endless debate-- basically, start-at-1 is a mathematician's preference, but it screws up signal processing calculations, particularly fourier transforms :( .

Optimizing loops: the current version of Matlab does a lot of loop optimization automagically, so a lot of the need to condense multiple loops in programs into clever one-liners has become unnecessary. That being said, it's always seemed to me that Matlab's vectorization abilities are more powerful than IDL's. Although maybe it's just a question of familiarity

> I run into lots of Matlab "functions" that I don't know.  
> Matlab has a \*wonderful\* web site that explains what these  
> things do. That is very helpful.

Syntax aside, the main chore in using both programs is remembering which of the thousand or so functions in the libraries to use for any given problem, so I'm a heavy user of the help facilities. Generally, I've found Matlab is better.

> In this application I'm seeing a lot of matrix algebra  
> that I have mostly forgotten. There are a lot of diagonals  
> being calculated, then a matrix being multiplied with something  
> else. I guess I don't do much math in IDL, or something, but  
> it has been a long time since I thought much about kind of  
> thing. The Matlab folks like to shift a power spectrum by  
> multiplying every other row and column by -1. I think a  
> simple SHIFT does the job very nicely, thank you!

Cleve Moler, the 'guru' who guided Matlab's early (and, I suspect, continues to influence current) development is an applied mathematician, so Matlab has a complete implementation of the standard linear algebra and special function libraries. On the other hand, if I want to read a GEOTIFF in Matlab, I'm out of luck.

> My main problem at the moment is that I have a hard time  
> seeing how these m files are put together. I don't really  
> see the "thread" of the application yet. There are a lot  
> of "global" variables that are important to me, but I have  
> not been able to see where these get their default or  
> starting values. I see the main m module that gets  
> everything started, but I don't see how initial  
> variable values are assigned. I don't see any  
> "optional" parameters in the function definitions.  
> (This could all be a function of the person writing  
> the code. Lord knows I've seen strange IDL programs  
> in my day!)

I suggest you use the debugger and watch when variables appear in the workspace.

> I guess my general impression is that Matlab looks  
> WAY more mathematically sophisticated than IDL, but  
> IDL looks WAY easier to program. I guess it was ever  
> thus. :-)

No question that IDL is more sophisticated on programming. Matlab lacks real pointers or 'heap' variables, so some things (like object orientation) have to be faked.

> Cheers,  
>  
> David

Matt Feinstein

--

There is no virtue in believing something that can be proved to be true.

---