Subject: Re: rotating 3d array: help for the braindead! Posted by thompson on Wed, 01 Mar 1995 17:01:12 GMT

View Forum Message <> Reply to Message

piclinch@dux.dundee.ac.uk (Pete Clinch) writes:

- > MEC (mark_cadwell@gmail4.sp.trw.com) wrote:
- > : I previously wrote:
- >: > I have a stack of MRI images that make up a 3d array, and I want to turn
- > : > all the constituent images over.
- >: > At the moment, I've got a FOR loop that goes through using
- > : > ROTATE(image(*,*,i),2)
- >:>
- >: > on each one, but I'm convinced there must be a better way to do it in a
- >: > oner without the FOR loop to slow things down.
- >: There is a method of animation called double buffering where you plot an
- > : image to a virtual window, copy the virtual window to your main window,
- > : and while that is going on, you're drawing the next image to the virtual
- > : window again. The advantage of doing it this way is that plotting to
- > : virtual windows is much faster (at least in my machine) that plotting to
- > : visible windows, and copying from a virtual window to a visible window is
- > : also much faster that plotting. The result is a nice, smooth, and
- > : reasonably fast image rotation.
- > I'm not actually after an animated rotation here... My image stack is
- > just plain upside down, and I want to flip each image in the stack over
- > *before* I actually start looking at them.
- > The image stacks are 128 images, 256 pixels square each: takes quite a
- > bit of time to flip them all :-(
- > Pete.
- > Peter Clinch **Dundee Teaching Hospitals NHS Trust**
- > voice: 44 1382 660111 x 3637 snail: Directorate of Medical Physics
- > fax: 44 1382 640177 Ninewells Hospital
- Dundee DD1 9SY Scotland UK > email: p.j.clinch@dundee.ac.uk

I have a routine called REARRANGE which should do what you want. It's basically a multi-dimensional equivalent of ROTATE, but with a somewhat different interface. You can find it at URL

file://umbra.gsfc.nasa.gov/pub/soho/soft/cds/util/array/rear range.pro

However, in order to make it work efficiently you will also need some CALL EXTERNAL software to support it. You can find this software (written in C) at URL

file://umbra.gsfc.nasa.gov/pub/soho/soft/cds/external

This C software is strictly needed, the routine is capable of working without it. However, it's much slower without the CALL_EXTERNAL software--probably about the same as what you're doing now.

Bill Thompson