
Subject: Large Image Handling FAQ?

Posted by [Andy Sowter](#) on Wed, 10 Mar 1999 08:00:00 GMT

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

I'm dealing with many large (130MB is a typical size - 16-bit integers) images at a time, doing matching and 3-d reconstruction. Are there any FAQs or links out there relating to tips to do this - I don't like just ramping up my RAM or virtual memory every time because, well, it doesn't feel right (and it only seems to cause problems in the long run).?

Thanks in advance

Andy

Subject: Re: Large Image Handling FAQ?

Posted by [paulwomack](#) on Thu, 11 Mar 1999 08:00:00 GMT

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

In article <7c5hu7\$23\$1@news7.svr.pol.co.uk>,

"Andy Sowter" <asowter@synopticsga.freemove.co.uk> wrote:

- > I'm dealing with many large (130MB is a typical size - 16-bit integers)
- > images at a time, doing matching and 3-d reconstruction. Are there any FAQs
- > or links out there relating to tips to do this - I don't like just ramping
- > up my RAM or virtual memory every time because, well, it doesn't feel right
- > (and it only seems to cause problems in the long run).?
- >
- > Thanks in advance
- >
- > Andy
- >
- >

OK. Assumption: you do not have enough RAM to have all your images (input(s) output(s) scratch(es)) in RAM simultaneously.

Trivial conclusion. You can only have PART of your images in RAM at any one time.

The only question is .. how?

The actual question is "what shape". The only 2 realistic answers are "stripes" i.e. set of full width horizontal lines.
"tiles" i.e. set of part width horizontal lines.

Stripes are good for "streaming" processing. You can view them as an extension of ordinary buffered IO. If you only move "forward" in an image, they are efficient, and easy(ish) to

implement.

"tiles" are what PhotoShop and GIMP use. The upshot is that (given tile size K), a square KxK pixels in stored in contiguous memory.

The IMPORTANT thing about tiles

is that nearby pixels are (in x,y space) are nearby in RAM. The key thing here is that in a "simple" raster moving from (x,y) to (x, y+1) has a nasty habit of paging in entire scan lines, even if you only want one pixel.

However tiles are not good for streaming. What you need is a uncompressed raster on the disc, that can be accessed by lseek/read combinations.

In fact, if you arrange your image in a "tile" system, your disc can be simply mmap(), and the system WON'T THRASH.

All image processing code has to access pixel(x,y) via a function that tests for tile boundaries. Alternatively for time critical code you can write function that work within tiles, so that the boundary code is handled as a separate case.

BugBear.

-----== Posted via Deja News, The Discussion Network ==-----

<http://www.dejanews.com/> Search, Read, Discuss, or Start Your Own
