Subject: Re: upsampling images
Posted by Klemen on Tue, 04 Jun 2013 22:00:38 GMT
View Forum Message <> Reply to Message

If you know, the number of cols /lins and you know how the pixels are positioned, you can use just interpolate function. But if you know, that each upscaled pixel should consist of e.g. 5 by 5 original pixels, then this example (upsample the MODIS from original swath to 5 by 5 averaged pixels). The whole code is at another thread:
https://groups.google.com/forum/?hl=en&fromgroups#!topic /comp.lang.idl-pvwave/aI6Vh_op7Lc

Cheers, Klemen

```
; Average original data to "geolocation frame"
	;first prepare indexes
	out_size = size(m_lat)                          ;the output will have a reduced spatial
resolution (corresponding to the geolocation)
	;the position 0,0 in geolocation corresponds to pixel 2,2 in original
data
	;the geolocation is 5 times downsampled
	out_indx_col = indgen(out_size[1]) * 5L + 2L        ;corresponding coloumns
of orig. data in downsampled grid
	out_indx_lin = indgen(out_size[2]) * 5L + 2L        ;corresponding lines of
orig. data in downsampled grid
	out_indx_col = rebin(out_indx_col, out_size[1], out_size[2])
	out_indx_lin = rebin(reform(out_indx_lin,1,out_size[2]), out_size[1],
out_size[2])
	out_indx = out_indx_lin * in_size[1] + out_indx_col        ;one dimensional
index of original data in downsampled grid

- hide quoted text -
	;compute mean value for the radiance
	m_count = make_array(out_size[1],out_size[2])              ;array containing the
number of good maeasurements
	m_mean = make_array(out_size[1],out_size[2])               ;array containing the
mean maeasurements
	for j=-2,2 do begin
		for i=-2,2 do begin
			indx = out_indx + out_size[1]*j + i
			tmp = m_modis[out_indx]
			indx_good = where(tmp le 32767)                    ;do not use nodata, etc.
			m_count[indx_good] = m_count[indx_good]  + 1
			m_mean[indx_good] = m_mean[indx_good] + tmp[indx_good]
		endfor
	endfor
	m_mean = m_mean / m_count
```