Subject: Re: Array manipulation
Posted by majewski on Sat, 04 Nov 2000 04:28:22 GMT
View Forum Message <> Reply to Message

Thanks Jaco (& everyone else) this works very well for my purposes (much faster than my for loop stuff).
I'll look into the median aspect latter

leon

On Wed, 1 Nov 2000 17:59:16 -0000, "Jaco van Gorkom" <gorkom@rijnh.nl> wrote:
> <snip>
> For just NaN-correction the code would look
> something like this:
>
> ArrDims = size(InputArr, /dimensions)
> CorrArr = $
> rebin(float(finite(InputArr)), ArrDims[0]/Width, ArrDims[1]/Width)
> InputArr[where(finite(InputArr, /NaN))] = 0
> Average = rebin(InputArr, ArrDims[0]/Width, ArrDims[1]/Width)
> CorrArr[where(CorrArr eq 0.)] = !values.f_nan ; to avoid division by zero
> Average = Average / CorrArr
>
> I don't know if this will still be faster than smart subscripting in a loop,
> it probably depends on the size of the arrays. It also uses somewhat more
> memory. I hope that someone else knows a better solution. It would be best
> for this type of NaN-handling to be a built-in option of IDL-routines like
> rebin, smooth, etc. I would certainly be using it!
> <snip>
> Jaco
>
> ----------
> Jaco van Gorkom
> FOM-Instituut voor Plasmafysica "Rijnhuizen", The Netherlands
> e-mail: gorkom@rijnh.nl
>

Leon Majewski

Remote Sensing & Satellite Research Group
Curtin University of Technology, Perth, Australia

email: majewski@ses.curtin.edu.au