Subject: Re: How to create a 2D mask that automatically half's an irregularly shaped 2D array from top to bottom?
Posted by Jeremy Bailin on Fri, 18 Nov 2011 21:59:20 GMT
View Forum Message <> Reply to Message

On 11/18/11 2:53 PM, Jeremy Bailin wrote:
> On 11/18/11 10:04 AM, Dr G. wrote:
>> Hi Folks,
>>
>> Q: Can the IDL geometry geniuses out there think of a fast way to
>> create a 2D mask that automatically halfï¿½s an irregularly shaped 2D
>> array along its x axis (i.e., from top to bottom) Eg:
>>
>> [0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0]
>> [0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0]
>> [0,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0]
>> [0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0]
>> [0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0]
>> [0,0,0,0,0,0,0,0,1,1,1,1,1,1,0,0]
>> [0,0,0,0,0,0,0,1,1,1,1,1,1,0,0,0]
>> [0,0,1,1,1,1,1,1,1,1,0,0,0,0,0,0]
>> [0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0]
>> [0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0]
>> [0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0]
>> [0,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0]
>>
>> Merci.
>>
>> Gf
>
> If the input mask is "inmask":
>
> ; how many 1s are there?
> rowtot = total(inmask, 1, /int)
> ; and it by checking if the cumulative total along the row is less
> ; than half of rowtot
> outmask = inmask and (total(inmask, 1, /int, /cumul) le $
> rebin(transpose(rowtot/2), masksize, /sample))
>
> -Jeremy.

Oops, missed the first line when I copied that in:

masksize = size(inmask, /dimen)

-Jeremy.