
Subject: Re: Convolution with non-constant Kernel?
Posted by [rogass](#) on Fri, 19 Nov 2010 10:07:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi, try the following routines for 2D images and 2D mask or mask sized [maskx,masky,n_elements(image)], whereas boundary pixels are not considered, large images and small masks will cause heavy memory usage:

```
function cr_discrete_convolve,mask,im
szm = ulong(size(mask,/dimensions))
szi = ulong(size(im,/dimensions))
return, reform(/
over,total(total(im[cr_window(szi[0],szi[1],szm[0],szm[1]])* $
  n_elements(szm) eq 2? rebin(/
sample,mask,szm[0],szm[1],szi[0]*szi[1]):$
  mask,2),1),szi[0],szi[1])
end

function cr_window,sx,sy,wx,wy,norot=norot
wx = ulong(wx)
wy = ulong(wy)
sx = ulong(sx)
sy = ulong(sy)
base = ((ll=ulindgen(wx,wy) mod wx)) + transpose(ll)*sx
return, rebin(/sample,(keyword_set(norot)? temporary(base) : $
  rot(reform(/over,temporary(base),wx,wy,180)),wx,wy,sx*sy) + $
  rebin(/sample,ulindgen(1,1,sx*sy),wx,wy,sx*sy)
end
```

Cheers

CR
