Subject: Re: Kuwahara Filter
Posted by Joshua Nipper on Thu, 14 Feb 2002 23:15:15 GMT
View Forum Message <> Reply to Message

That was an old version of the code, this is what I'm currently using....
Josh

```
function kuwahara,input,kernelsize
T=systime(1)
s=size(input,/dimensions)
temp=bytarr(s)
identity=intarr(kernelsize)+1
M=indgen(kernelsize)-(kernelsize/2)
mX=M#identity
my=-identity#M
for i=0,s[0]-1 do begin
 for j=0,s[1]-1 do begin
  xpoint=j
  ypoint=i
  region1=input[(xpoint-1)+mX,(ypoint+1)+mY]
  region2=input[(xpoint+1)+mX,(ypoint+1)+mY]
  region3=input[(xpoint-1)+mX,(ypoint-1)+mY]
  region4=input[(xpoint+1)+mX,(ypoint-1)+mY]

 var=[variance(region1),variance(region2),variance(region3),v ariance(region4)
]
   avg=fix([mean(region1),mean(region2),mean(region3),mean(regi on4)])
  location=where(var EQ min(var))
  temp[xpoint,ypoint]=avg[location[0]]
 endfor
endfor
PRINT, SYSTIME(1) - T, 'Seconds'
return,temp

end
```

Subject: Re: Kuwahara Filter
Posted by Jaco van Gorkom on Fri, 15 Feb 2002 10:57:31 GMT
View Forum Message <> Reply to Message

"Joshua Nipper" <nipperjc@ufl.edu> wrote in message
news:a4hcm8piu1@spnode25.nerdc.ufl.edu...
> Hi there, I'm trying to implement the kuwahara filter, and can't seem to
> figure out how to get around using for loops, making it very slow. Can
> anyone suggest a way to speed this up?
< code skipped >

Without being aware of the finer details of the kawahara filter, I think
that you need the variance for neighbourhoods of kernelsize around
each point in your data, and then based on the variances you find around
each input point, you choose the mean value of one of the neighbourhoods.
So how about calculating the mean values for *all* neighbourhoods first,
loopless:
  means = smooth(input, kernelsize)
and then calculating the variances for all neighbourhoods, again loopless:
  variances = smooth((input - means)^2, kernelsize)
followed by some clever selection step output = means[cleverStep(variances)]
involving possibly the < operator, MIN(), SHIFT() or what may be.
If all else fails, use a loop for this last step.

Hope this helps,
  Jaco
