Subject: Calculate maximum diameter?
Posted by Jeff N. on Sat, 04 Feb 2006 20:28:07 GMT
View Forum Message <> Reply to Message

Hi folks,

I have a set of roughly circular ROI's for which I want to compute the maximum diameter - the longest straight line from one point on the boundary of the ROI, through the center, to the opposing boundary. Any ideas about the best way to do this? I do a loop I guess....compute the diameter for all 360 degrees around the center. Principal components analysis also springs to mind, but some of these ROI's have different scales in the x and y dimensions. Can anyone help?

Jeff

Subject: Re: Calculate maximum diameter?
Posted by greg michael on Mon, 06 Feb 2006 13:14:51 GMT
View Forum Message <> Reply to Message

It depends on what you mean by 'roughly' - if the radius is a random function, you won't do any better than to search them all; if you means an ellipse, or some other predictable deviation from circular, you could, of course, get there quicker.

Greg

Subject: Re: Calculate maximum diameter?
Posted by James Kuyper on Mon, 06 Feb 2006 15:41:43 GMT
View Forum Message <> Reply to Message

Jeff N. wrote:
> Hi folks,
>
> I have a set of roughly circular ROI's for which I want to compute the
> maximum diameter - the longest straight line from one point on the
> boundary of the ROI, through the center, to the opposing boundary.

Is it essential that it go through the center? In many contexts its more useful to know the length of the longest straight line from any given vertex to any other vertex. There's a pretty obvious O(N^2) algorithm for finding that maximun distance.

Subject: Re: Calculate maximum diameter?
Posted by Jeff N. on Mon, 06 Feb 2006 22:36:24 GMT
View Forum Message <> Reply to Message

Yep, it does need to go through the center. I may end up using a
fitted ellipse as a starting point, so we'll just see what works best.
Thanks for the ideas!

Jeff

kuyper@wizard.net wrote:
> Jeff N. wrote:
>> Hi folks,
>>
>> I have a set of roughly circular ROI's for which I want to compute the
>> maximum diameter - the longest straight line from one point on the
>> boundary of the ROI, through the center, to the opposing boundary.
>
> Is it essential that it go through the center? In many contexts its
> more useful to know the length of the longest straight line from any
> given vertex to any other vertex. There's a pretty obvious O(N^2)
> algorithm for finding that maximun distance.

Subject: Re: Calculate maximum diameter?
Posted by Paul Van Delst[1] on Mon, 06 Feb 2006 23:17:10 GMT
View Forum Message <> Reply to Message

Jeff N. wrote:
> Yep, it does need to go through the center. I may end up using a
> fitted ellipse as a starting point, so we'll just see what works best.
> Thanks for the ideas!

How do you define the center of your "roughly circular" ROI's? Geometric centroid? Does this have to correspond to a discrete ROI pixel? And how do you define diameter? From one ROI pixel through the "center" to another ROI pixel? Or can the centroid and the opposite side pixel be non-discrete? Depending on how "big" a pixel is compared to your length units, maybe it doesn't matter (e.g. if a length unit was a pixel)

After computing your centroid, why can't you just rotate through 180deg. from one ROI pixel to the next, construct a line from your current pixel through your centroid and determine where it intersects your ROI on the other side? It seems a bit ham-fisted maybe, but it doesn't rely on any particular definition of "roughly circular."

I'm curious as to why you need the maximum diameter of your ROI's. If you have a lot of them, usually a particular orientation is chosen -- with a common defined axis of symmetry assumed to be random with respect to that orientation -- and the dimension of your ROI's

along that orientation is used (e.g. to compute statistics on particle sizes).

paulv

>
> Jeff
>
>
> kuyper@wizard.net wrote:
>
>> Jeff N. wrote:
>>
>>> Hi folks,
>>>
>>> I have a set of roughly circular ROI's for which I want to compute the
>>> maximum diameter - the longest straight line from one point on the
>>> boundary of the ROI, through the center, to the opposing boundary.
>>
>> Is it essential that it go through the center? In many contexts its
>> more useful to know the length of the longest straight line from any
>> given vertex to any other vertex. There's a pretty obvious O(N^2)
>> algorithm for finding that maximun distance.
>
>

--
Paul van Delst
CIMSS @ NOAA/NCEP/EMC

Subject: Re: Calculate maximum diameter?
Posted by Karsten Rodenacker on Tue, 07 Feb 2006 08:06:11 GMT
View Forum Message <> Reply to Message

Hmm, possibly the attached routine is of some help. It is extracted from a
thesis http://www.isical.ac.in/~amitava/thesis.ps
Regards
Karsten

Am Sat, 04 Feb 2006 21:28:07 +0100 schrieb Jeff N. <jnettle1@utk.edu>:

> Hi folks,
>
> I have a set of roughly circular ROI's for which I want to compute the
> maximum diameter - the longest straight line from one point on the
> boundary of the ROI, through the center, to the opposing boundary. Any

> ideas about the best way to do this? I do a loop I guess....compute
> the diameter for all 360 degrees around the center. Principal
> components analysis also springs to mind, but some of these ROI's have
> different scales in the x and y dimensions. Can anyone help?
>
> Jeff
>

```
; $Id: ad_msc.pro,v 1.2 2006/02/07 08:04:03 karo Exp $
;+
; NAME:
; AD_MSC
;
; PURPOSE:
; Calculation of the minimumspanning circle
;
; CATEGORY:
; Geometry
;
; CALLING SEQUENCE:
; Rad=ad_msc(X,Alpha)
;
; INPUTS:
; X: 2/3 dim Vector with coordinates [3,nx]
;   Alpha: Weight
;
; KEYWORD PARAMETERS:
; WEIGHT: Switch set decreased weight, def. constant weight
;
; MAX_ITERATIONS: Number of iterations
;   CENTER: variable, will contain the coordinates of msc
;   EPS: Epsilon for exact match
;   NP: Number of nonprinted lines if PRINT is on
;   PRINT: Switch for printing
;
; OUTPUTS:
; Result: Radius of MSC
;
; PROCEDURE:
;
; EXAMPLE:
; See in program file
;
;
; MODIFICATION HISTORY:
;  Written by: Karsten Rodenacker, after A. Datta, 1998.
; Nov. 99 Revised for 3d
;       $Log: ad_msc.pro,v $
```

```
;       Revision 1.2  2006/02/07 08:04:03  karo
;       Conversion Win to Mac
;
;
  FUNCTION ad_msc, x, a, $
                MAX_ITERATIONS=nit, WEIGHT=ia, $
                CENTER=cent,EPS=eps, NP=np, $
                PRINT=print,HIT_POINT=i1
;-
    IF NOT keyword_set(eps) THEN eps=0.1E-7
    eps=double(eps)
;  if not keyword_set(nit) then nit=2000l
    IF NOT keyword_set(nit) THEN nit=(size(x,/dim))[1]*10.
    IF NOT keyword_set(np) THEN np=100
    sx=size(x)
    nx=n_elements(x)/sx[1]

    w=double(total(x[0,*])/nx)
    FOR i=1,sx[1]-1 DO w=[temporary(w),double(total(x[i,*])/nx)]
    it=0l
    CASE sx[1] OF
       2: REPEAT BEGIN
          IF NOT keyword_set(ia) THEN alpha=double(a) $
          ELSE alpha=double(a/(it+1))
          mxd=max(sqrt((x[0,*]-w[0])^2+(x[1,*]-w[1])^2), i1)
          pw=w
          w=w+alpha*(x[*,i1]-w)
          it=it+1
          IF keyword_set(print) AND (it MOD np) THEN $
            print,eps, ABS(pw-w)
       ENDREP UNTIL (max(abs(pw-w)) LT eps) OR (it GE nit)
       3: REPEAT BEGIN
          IF NOT keyword_set(ia) THEN alpha=double(a) $
          ELSE alpha=double(a/(it+1))
           mxd=max(sqrt((x[0,*]-w[0])^2+(x[1,*]-w[1])^2+(x[2,*]-w[2])^2 ),
i1)
          pw=w
          w=w+alpha*(x[*,i1]-w)
          it=it+1
          IF keyword_set(print) AND (it MOD np) THEN $
            print,eps, ABS(pw-w)
       ENDREP UNTIL (max(abs(pw-w)) LT eps) OR (it GE nit)
       ELSE: message,'dimension not implemented'
    ENDCASE
    cent=w
;  return, sqrt((x[0,i1]-w[0])^2+(x[1,i1]-w[1])^2)
    return, sqrt(total((x[*,i1]-w)^2))
END
```

```
; test wird nur bei .run ad_msc aufgerufen
   x=randomn(seed,2,100)
   rad=ad_msc(x,0.0001d,cent=c)
   ci=ro_circle(c[0],c[1],rad)
   window,0
   plot,ci[0,*],ci[1,*]
   plots,c[0],c[1],psym=2
   oplot,x[0,*],x[1,*],/psym
   x=randomn(seed,3,100)
   rad=ad_msc(x,0.0001d,cent=c)
   ci=ro_circle(c[0],c[1],rad)
   window,1
   plot_3dbox,x[0,*],x[1,*],x[2,*],psym=4,/solid
   plots,ci[0,*],ci[1,*],replicate(c[2],32),col=0,/t3d
   ci=ro_circle(c[1],c[2],rad)
   plots,replicate(c[0],32),ci[0,*],ci[1,*],col=0,/t3d
   ci=ro_circle(c[0],c[2],rad)
   plots,ci[0,*],replicate(c[1],32),ci[1,*],col=0,/t3d
    plots,[c[0]-rad,c[0]+rad],[c[1],c[1]],[c[2],c[2]],col=0,/t3d
    plots,[c[0],c[0]],[c[1]-rad,c[1]+rad],[c[2],c[2]],col=0,/t3d
    plots,[c[0],c[0]],[c[1],c[1]],[c[2]-rad,c[2]+rad],col=0,/t3d
   wshow
   !p.multi=0
END
```

--
Erstellt mit Operas revolutionï¿œrem E-Mail-Modul: http://www.opera.com/m2/

Subject: Re: Calculate maximum diameter?
Posted by Jeff N. on Wed, 08 Feb 2006 23:01:36 GMT
View Forum Message <> Reply to Message

Thanks for the code Karsten, I'll give it a look!

Subject: Re: Calculate maximum diameter?
Posted by Jeff N. on Wed, 08 Feb 2006 23:18:05 GMT
View Forum Message <> Reply to Message

Paul Van Delst wrote:

> How do you define the center of your "roughly circular" ROI's? Geometric centroid? Does
> this have to correspond to a discrete ROI pixel? And how do you define diameter? From one
> ROI pixel through the "center" to another ROI pixel? Or can the centroid and the opposite
> side pixel be non-discrete? Depending on how "big" a pixel is compared to your length

> units, maybe it doesn't matter (e.g. if a length unit was a pixel)

Yes, I'm defining the center as the geometric centroid. I don't necessarily think that it has to correspond to a discrete pixel, but you are right that this probably doesn't matter, and you guessed the reason why too.

>
> After computing your centroid, why can't you just rotate through 180deg. from one ROI
> pixel to the next, construct a line from your current pixel through your centroid and
> determine where it intersects your ROI on the other side? It seems a bit ham-fisted maybe,
> but it doesn't rely on any particular definition of "roughly circular."

I can, and there's a good chance this is what I'll do. I'm trying to decide if the PCA would be more effective though. Plus, I'm going to probably have trouble coding this approach :) I understand that how i'm defining "roughly circular" is important....I think that they're circular enough (not shaped like stars or anything) so that the PCA would work. If this helps, I've done some measurements on these that are the perimeter of the ROI divided by the perimeter of a circle with the same area as the ROI, and these measurements are usually within 30-40% of a circle. (why am i using that kind of measurement? b/c someone else used that method and i'm evaluating it :))

>
> I'm curious as to why you need the maximum diameter of your ROI's. If you have a lot of
> them, usually a particular orientation is chosen -- with a common defined axis of symmetry
> assumed to be random with respect to that orientation -- and the dimension of your ROI's
> along that orientation is used (e.g. to compute statistics on particle sizes).
>
The ROI's are outlines of mineral groups in meteorites (they're chondrules if you're familiar with meteorites). Measurements of this sort are "standards" in meteorite literature - everyone else uses them, so if i'm going to compare my work to anyone else's I've got to use these.

Thanks for your help!
Jeff