

---

Subject: MORPH\_XYZ and values keyword  
Posted by [btt](#) on Tue, 24 Jun 2003 21:00:40 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello,

I completely mystified by the VALUES keyword to the MORPH\_XYZ (like MORPH\_TOPHAT) routines. I think I understand the meaning of VALUES for DILATE and ERODE-like routines - but I can't make sense of the MORPH\_XYZ routines.

Here's the description of the so-called structuring element that these routine accept...it gives the shape and 'on-off'-ness of each pixel.

A one-, two-, or three-dimensional array to be used as the structuring element. The elements are interpreted as binary values  $i_j$  either zero or nonzero. The structuring element must have the same number of dimensions as the Image argument.

Now here's the decription of the VALUES keyword...

An array of the same dimensions as the Structure argument providing the values of the structuring element. If the VALUES keyword is not present, all elements of the structuring element are 0.

What the heck does that mean?

I thought providing the VALUES was like providing a weighting to the structuring element.

Here's an example using the MORPH\_TOPHAT routine... if here's a structure with a narrow brim - any peaks detected that fit within the 1s is a keeper.

```
0 0 0 1 0 0 0
0 1 1 1 1 1 0
0 1 1 1 1 1 0
1 1 1 1 1 1 1
0 1 1 1 1 1 0
0 1 1 1 1 1 0
0 0 0 1 0 0 0
```

Now if I am working with a grayscale image... I want to make sure the peak is tall (bright) enough so, I weight the structuring element with

some kind of a height - like I want the peak to be at least 5 whatever  
above the brim values. So I set VALUES equal to...

```
0 0 0 5 0 0 0
0 5 5 5 5 5 0
0 5 5 5 5 5 0
5 5 5 5 5 5 5
0 5 5 5 5 5 0
0 5 5 5 5 5 0
0 0 0 5 0 0 0
```

I can't see any difference between the results when I do and do not  
provide the VALUES. Which, of course, means I don't understand it at all.

Stumped and grateful for any help,

Ben

---

Subject: Re: MORPH\_XYZ and values keyword  
Posted by [Karsten Rodenacker](#) on Thu, 26 Jun 2003 06:24:13 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Hi, I have slightly changed your Morphtest example into Morphtest1 with  
two parameters, extension and type. You should play around with  
extension and gen\_sphere type. You will see, that the flat value  
structuring element (STE) does not show considerable differences to the  
simple STE. However the last example shows some influences of value STE.

Flat STE controls local extension, value STE controls value extension  
(and function shape).

To learn to understand morphological operations morph\_tophat is IMHO not  
the best routine. Since it is simply the differenc between original and  
opened array, I have replaced it by open and close, the ingredients for  
bright and dark tophat.

Simply run Morphtest1 will open 4 windows with different parameters  
showing opening and closing.

Regards  
Karsten

Ben Tupper schrieb:

> Karsten Rodenacker wrote:

>

>> Hallo, maybe some information concerning VALUE keywords. Let ste the

```

>> structuring element and val the values array.
>> 1. Only val*(ste ne 0) is of importance!
>> 2. As far as I know are the val values subtracted or added from the
>> input data before the max or min according erode or dilate is
>> performed. Hence, flat value arrays do not make sense.
>> 3. I attach two routines (for 1-d morphological filtering) for
>> illustration. gen_sphere to generate the value array and mm_filter to
>> apply a sequence of increasing filter steps. Unluckily the routines
>> are not documented. Maybe they are comprehensible without further words.
>>
>
>
> Thanks Karsten,
>
> This helps very much. I'm still not clear on why a flat-topped VALUES
> do not make sense. Here's what I get from the online docs for ERODE
> and DILATE. They don't look symmetric to me - so it seems like a
> flat-topped kernel would have a meaningful effect. But, I'm new at this
> and feeling like I'm on pretty thin ice.
>
>
> Erode... result = min(Image) - VALUE
> "Each pixel of the result is the minimum of Image less the corresponding
> elements of VALUE."
>
> Dilate... result = max(Image + VALUE)
> "Each pixel of the result is the maximum of the sum of the corresponding
> elements of VALUE and the Image pixel value. "
>
> The Tophat filter is supposed to be...
> result = (Erosion followed by Dilation) - OriginalImage
>
> I have mocked up an 1-d example (below) using your gen_sphere routine.
> It looks to me like I might be better off *not* using the VALUES anyway.
>
> Just type morphtest from the command line. The top plot is the tophat
> without values specified. The second is with the fat-topped (the VALUES
> structure is shown hovering over the plot on the right.) And the last
> is using a non-fat-topped structure (also shown at hovering at right.)
>
> Thanks,
> Ben
>
>
> *****begin code
> function gen_sphere, rad, TYPE=typ
>
> if not keyword_set(typ) then typ=0

```

```

> case typ of
>   1:v = sqrt(rad^2-(rad-findgen(rad*2+1))^2)
>   2:v = rad^2-(rad-findgen(rad*2+1))^2
>   3:v = 0.05/(1.+sqrt(rad^2-(rad-findgen(rad*2+1))^2))
>   4:v = 0.05/(1.+rad^2-(rad-findgen(rad*2+1))^2)
>   5:v = rad-sqrt(rad^2-(rad-findgen(rad*2+1))^2)
>   6:v = rad^2-(rad-findgen(rad*2+1))^2
>   -1:v = replicate(1.0,rad*2+1)
>   else: v = rad-abs(-findgen(rad*2+1)+rad)
> ENDCASE
> mv = max(v)
> IF mv GT (rad+1) THEN v = v/mv*(rad+1)
> return, v
> end
>
>
>
> Pro MorphTest
>
> x = [19,19,21,23,24,23,22,21,20,19,20,20,19,17,15,$
> 13,11,11,12,14,17,21,24,25,24,23,21,21,23,26,29,30,27,$
> 22,17,12,10,10,12,14,16,19,20,21,20,17,12,8,$
> 4,3,3,6,10,14,19,23,26,27,28,29,29,29,27,$
> 23,19,16,14,14,16,18,19,19,18,15,13,11,7,$
> 4,1,0,0,3,6,10,12,12,11,8,5,2,1,1,4,7,11,14,17,20,23,27]
>
>
> n = 10
>
> old_p = !P
> !P.CharSize = 2
> !P.Multi = [0,1,3]
>
> Dim = get_Screen_Size()
> Window, /Free, ys = dim[1]*0.9
> plot, x, title = 'orig with simple tophat'
> oplot, thick = 2, $
>   morph_tophat(x, [0, replicate(1,n-1), 0])
>
> plot, x, title = 'orig with tophat values set with flat top'
> values = [0, replicate(9,n-1),0]
> oplot, thick = 2, $
>   morph_tophat(x, [0, replicate(1,n-1), 0], values = values )
> oplot, Indgen(n+1) + 80, values + 10, thick = 2
>
>
> plot, x, title = 'orig with tophat values set with rounded top'

```

```

> values = gen_sphere(n/2)
> oplot, thick =2, $
>   morph_tophat(x, [0, replicate(1,n-1), 0],values =values )
> oplot, Indgen(n+1) + 80, values + 10, thick = 2
>
> !P = old_p
> End
> *****end code
>

```

Pro MorphTest1, n, t

```

x = [19,19,21,23,24,23,22,21,20,19,20,20,19,17,15,$
13,11,11,12,14,17,21,24,25,24,23,21,21,23,26,29,30,27,$
22,17,12,10,10,12,14,16,19,20,21,20,17,12,8,$
4,3,3,6,10,14,19,23,26,27,28,29,29,29,27,$
23,19,16,14,14,16,18,19,19,18,15,13,11,7,$
4,1,0,0,3,6,10,12,12,11,8,5,2,1,1,4,7,11,14,17,20,23,27]

```

```

if n_params() eq 0 then n = 10
if n_params() le 1 then t = 1

```

```

old_p = !P
!P.CharSize = 2
!P.Multi = [0,1,3]

```

```

Dim = get_Screen_Size()
Window, /Free, ys = dim[1]*0.9
plot, x, title = 'orig with simple open/close'
oplot, thick = 2, $
  morph_open(x, [0b,replicate(1b,n-1),0b],/gray)
oplot, thick = 1, linest=1, $
  morph_close(x, [0b,replicate(1b,n-1),0b],/gray)

```

```

plot, x, title = 'orig with open/close values set flat top'
values = [0b, replicate(byte(n-1),n-1),0b]
oplot, thick = 2, $
  morph_open(x, [0b, replicate(1b,n-1), 0b], values = values )
oplot, thick = 1,linest=1, $
  morph_close(x, [0b, replicate(1b,n-1), 0b], values = values )
oplot, Indgen(n+1) + 80, values + 10, thick = 2

```

```

plot, x, title = 'orig with open/close values set with gen_sphere'
values = gen_sphere(n/2, typ=t)
oplot, thick =2, $
  morph_open(x, [0b, replicate(1b,n-1), 0b],values =values )

```

```
oplot, thick =1, linestyle=1, $  
  morph_close(x, [0b, replicate(1b,n-1), 0b],values =values )  
oplot, Indgen(n+1) + 80, values + 10, thick = 2
```

```
!P = old_p  
End  
for i=6,12,2 do morphtest1,i,1  
end
```

## File Attachments

---

1) [MorphTest1.pro](#), downloaded 102 times

---

---

Subject: Re: MORPH\_XYZ and values keyword  
Posted by [btt](#) on Thu, 26 Jun 2003 17:13:05 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

In article <3EFA918D.2010108@gsf.de>,  
Karsten Rodenacker <Karsten.Rodenacker@gsf.de> wrote:

> Hi, I have slightly changed your Morphtest example into Morphtest1 with  
> two parameters, extension and type. You should play around with  
> extension and gen\_sphere type. You will see, that the flat value  
> structuring element (STE) does not show considerable differences to the  
> simple STE. However the last example shows some influences of value STE.  
>  
> Flat STE controls local extension, value STE controls value extension  
> (and function shape).  
>  
> To learn to understand morphological operations morph\_tophat is IMHO not  
> the best routine. Since it is simply the differenc between original and  
> opened array, I have replaced it by open and close, the ingredients for  
> bright and dark tophat.  
>  
> Simply run Morphtest1 will open 4 windows with different parameters  
> showing opening and closing.  
>

Hi,

I posted a message earlier today - but the pesky server is having a bad hair day. Thanks so much - the fog is a bit less dense now. The lower level routines do give a heck of a lot more control, as you show it.

I modified the code, yet again, to show how the various TOPHAT filter work. On each plot is shown the data (thin line), the opening (thick

line) and the tophat (dashed.) On each plot, RSI's MORPH\_TOPHAT is shown in the top plot, in the subsequent ones show

```
tophatResult = original - openedOriginal
```

using the various VALUES provided by your GEN\_SPHERE routine.

Thanks again,  
Ben

```
*****begin code  
Pro MorphTest, n, t
```

```
x = [19,19,21,23,24,23,22,21,20,19,20,20,19,17,15,$  
13,11,11,12,14,17,21,24,25,24,23,21,21,23,26,29,30,27,$  
22,17,12,10,10,12,14,16,19,20,21,20,17,12,8,$  
4,3,3,6,10,14,19,23,26,27,28,29,29,29,29,27,$  
23,19,16,14,14,16,18,19,19,18,15,13,11,7,$  
4,1,0,0,3,6,10,12,12,11,8,5,2,1,1,4,7,11,14,17,20,23,27]
```

```
if n_params() eq 0 then n = 10
```

```
if n_params() le 1 then t = 1
```

```
old_p = !P
```

```
!P.CharSize = 2
```

```
!P.Multi = [0,1,3]
```

```
Dim = get_Screen_Size()
```

```
Window, /Free, ys = dim[1]*0.9
```

```
plot, x, title = "orig with simple open/close"
```

```
oplot, thick = 2, $
```

```
    morph_open(x, [0b,replicate(1b,n-1),0b],/gray)
```

```
;oplot, thick = 1, linestyle=1, $
```

```
;    morph_close(x, [0b,replicate(1b,n-1),0b],/gray)
```

```
oplot, thick = 2, linestyle = 2, $
```

```
    morph_Tophat(x,[0b,replicate(1b,n-1),0b])
```

```
plot, x, title = 'orig with open/close values set flat top'
```

```
values = [0b, replicate(byte(n-1),n-1),0b]
```

```
op = morph_open(x, [0b, replicate(1b,n-1), 0b], values = values )
```

```
oplot, thick = 2, op
```

```
;oplot, thick = 1,linest=1, $
```

```
;    morph_close(x, [0b, replicate(1b,n-1), 0b], values = values )
```

```
oplot, thick = 2, linestyle = 2, x-op
```

```
oplot, Indgen(n+1) + 80, values + 10, thick = 2
```

```
plot, x, title = 'orig with open/close values set with gen_sphere'
```

```
values = gen_sphere(n/2, typ=t)
op = morph_open(x, [0b, replicate(1b,n-1), 0b], values = values )
oplot, thick =2, $
    morph_open(x, [0b, replicate(1b,n-1), 0b],values =values )
;oplot, thick =1, linestyle=1, $
; morph_close(x, [0b, replicate(1b,n-1), 0b],values =values )
oplot, thick = 2, linestyle = 2, x-op
oplot, Indgen(n+1) + 80, values + 10, thick = 2

!P = old_p
End
```

PRO morphtest1

```
for i=6,12,2 do morphtest,i,1
```

```
end
*****end code
```

```
--
Ben Tupper
Bigelow Laboratory for Ocean Sciences
btupper@bigelow.org
```

---