Subject: Re: Erosion/Dilation Operators
Posted by Achim Hein on Tue, 18 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Hi,
>
> Does anyone have any practical experience using the erosion
> and dilation operators in IDL? I can't make much sense of the
> documentation. In particular, I can't figure out what a good
> structuring element would be.
>
> I am working with ice flow data. I want to calculate the
> size of the ice flows, but first I have to pick them out.
> I've fooled around with thresholding, edge operators, etc.,
> but in the end the data is still "messy", without clearly
> defined boundaries, or with small "artifact" ice flows
> inside the larger, definate ice flows. I thought erosion
> and dilation would be a good way to clean up the data,
> but I can't see how it works exactly.
>
> I would appreciate any and all suggestions.
>
> Thanks,
>
> David
>

Hi David,

erode/dilate are morphological functions to find special areas due to a structuring element.
In principal we can say that for the erosion the algorithm looks if ALL pixels of an interesting area in the image are identical to the structuring element. If the pixels are equal to the structuring element the actual pixel in the resulting image is set to one.
The dilation looks if ONLY ONE pixel is equal to the structuring element before setting the resulting pixel to one.

I do not know any english literatur describing this problem exactly - sorry.

These functions are a very fast way to clean up data due to special expectation but think about the border of dilated/eroded areas - you shift the border of the eroded area and so you can lost information you are interested in.

If you are interested in ice flow, in the frequency domain the flowing areas had to be shifted as against the nonflowing parts?

Achim
--

Dipl.-Ing. A. Hein
PB2 / ZESS - Uni-GH-Siegen
Paul-Bonatz Str. 9-11
57068 Siegen
Phone: 0271/740-3362
Fax: 0271/740-2336

Please have a look at our Web-Sites:

http://www.nv.et-inf.uni-siegen.de/pb2/www_pb2

Subject: Re: Erosion/Dilation Operators
Posted by Kirt Schaper on Wed, 19 Feb 1997 08:00:00 GMT

View Forum Message <> Reply to Message

David Fanning wrote:
>
> Hi,
>
> Does anyone have any practical experience using the erosion
> and dilation operators in IDL? I can't make much sense of the
> documentation. In particular, I can't figure out what a good
> structuring element would be.

The way I think about it is that the structuring element (SE) is like a kernel for a convolution. it is centered over each pixel in the input image, and if all non-zero pixels in the SE are also non-zero in the input image, then the pixel in the output image is set.

The most common SE's are square (or circular) masks 'centered' in the middle, e.g. replicate(1,3,3) or:

```
se = bytarr(5,5)
for i=0,4 do &
  for j = 0,4 do &
    if (sqrt((i-2)^2 + (j-2)^2) le 2) then &
      se(i,j) = 1
```

Warning: non-symmetric and/or non-centered structuring elements
can lead to serious headaches.

```
-----------------------------------------------------
Kirt Schaper        | Bell: (612) 725-2000 x4791
PET Center (11P)    | net: kirt@pet.med.va.gov
VA Medical Center   | FAX: (612) 725-2068
MPLS, MN  55417     | URL: http://pet.med.va.gov:8080
```

Subject: Re: Erosion/Dilation Operators
Posted by djackson on Wed, 19 Feb 1997 08:00:00 GMT

View Forum Message <> Reply to Message

In article <davidf-ya023080001702972225400001@news.frii.com>,
davidf@dfanning.com (David Fanning) wrote:

> Does anyone have any practical experience using the erosion
> and dilation operators in IDL? I can't make much sense of the
> documentation. In particular, I can't figure out what a good
> structuring element would be.

Hi David,

I've found erosion and dilation to be useful in filtering out 'islands' in binary (thresholded) images that are smaller than what I'm interested in.
In general,

I've found a circular structuring element to be most useful, so as not to introduce a horizontal or vertical bias (boxy-ness!) to the filtering.

It took me a while to get the binary_circle function (for integer 'n' only, please!) to make nice-looking circles for me. See if this has any use for you:

```
---

function binary_circle, n
  dist1d = (findgen(n) - (n-1)/2.) ^ 2
  dist2d = sqrt(dist1d # replicate(1, n) + replicate(1, n) # dist1d)
  return, dist2d lt (n/2.)
end

function filter_it, image, n
  return, dilate(erode(image, binary_circle(n)), binary_circle(n))
end

---
```

You can test this with:

```
IDL> a=(randomu(seed,200,200) gt 0.25)  ; speckle field of mostly 1's
IDL> tvscl,a
IDL> tvscl,filter_it(a,4)
```

This should give you a lot of circles and joined paths at least as big as a circle of diameter 4.

If you want to filter out 0's and keep 1's, I believe you have to just invert the input array, do the filter_it, and invert the result.

Now, if you want to work with grayscale images and grayscale erode and dilate, you could replace binary_circle with gray_circle, and use the following, but I'm not sure if this is more helpful.

```
function gray_circle, n
  dist1d = (findgen(n) - (n-1)/2.) ^ 2
  dist2d = sqrt(dist1d # replicate(1, n) + replicate(1, n) # dist1d)
  return, ((((n/2.) - dist2d) / (n/2.)) > 0) * 2
end
```

Good luck, David! [I cringe at my own lack of comments... please be gentle!]

PS: I don't meaning to be a twerp, but isn't it "ice floes" when referring to floating masses of ice?

Cheers from the land of much ice!
-Dick

Dick Jackson   djackson@ibd.nrc.ca   Institute for Biodiagnostics
Opinions are mine alone.   National Research Council Canada, Winnipeg
   "And I told him my dream was to live for all time
      In some perfect refrain, like the man who wrote 'Danny Boy'."
        - Joe Jackson, from the album _Night_Music_, 1994.

Subject: Re: Erosion/Dilation Operators
Posted by Christian Soeller on Wed, 19 Feb 1997 08:00:00 GMT

View Forum Message <> Reply to Message

First of all, let me point out that I am surprised by the agressive style of your response to my followup (please point out to me what offended you in my posting). If you think that's the way to keep up the netiquette, but there you go...

> ???
> In your opinion, you do not loose information if you erase values from
> an array?

You do loose information by applying any non-invertable operation to an array.
I don't think that was the point of the original problem. Furthermore, it
is typical to 'deliberately' give up information when you do data reduction
(in this case e.g. the exact outline if you are mainly interested in features
on a larger scale (but this depends on what David really wants)).

>
>> You normally do an erosion (or several erosions) followed
>> by the same number of dilations with the same structuring elements.
>
> Normally means the way you use it?

I still keep up my point that this is a typical way of using dilations/
erosions when you want to remove small areas and smooth boundaries. Isn't
an erosion followed by a dilation related to the morphological "opening"
operation ?

> And what is smoothing with respect to the frequency domain?
> And where is the high frequency information after low-pass-filtering?
>

Well, this very much depends on the application, I am not sure if David wants
to keep the high frequency content (which is often related to the presence of
noise).

>
> Sorry for the lesson like Mail but what about 'Not nessesarily true'
>

Well, means in German "nicht unbedingt richtig" when the opening is e.g.
applied to objects with sufficiently smooth boundaries.

Hopefully we did not confuse David more than anything else. Anyway, to be able
to point out the best way to deal with the iceflow data some more detail about
what kind of data you are working on would be helpful, David.

Cheers,

Christian

-- --------
Christian Soeller mailto: csoelle@sghms.ac.uk

St. Georges Hospital Medical School   Dept. of Pharmacology
Cranmer Terrace   London SW17 0RE

Subject: Re: Erosion/Dilation Operators
Posted by Achim Hein on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Christian Soeller wrote:
>
>> These functions are a very fast way to clean up data due to special
>> expectation but think about the border of dilated/eroded areas - you
>> shift the border of the eroded area and so you can lost information you
>> are interested in.
>
> Not necessarily true.

???
In your opinion, you do not loose information if you erase values from
an array?

> You normally do an erosion (or several erosions) followed
> by the same number of dilations with the same structuring elements.

Normally means the way you use it?

> That will remove small areas and smooth boundaries of remaining areas >without shifting them too much.

And what is smoothing with respect to the frequency domain?
And where is the high frequency information after low-pass-filtering?

Sorry for the lesson like Mail but what about 'Not nessesarily true'

Achim
--

Dipl.-Ing. A. Hein
PB2 / ZESS - Uni-GH-Siegen
Paul-Bonatz Str. 9-11
57068 Siegen
Phone: 0271/740-3362
Fax: 0271/740-2336

Please have a look at our Web-Sites:

http://www.nv.et-inf.uni-siegen.de/pb2/www_pb2

Subject: Re: Erosion/Dilation Operators
Posted by Christian Soeller on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Achim Hein <hein@nv.et-inf.uni-siegen.de> writes:

> These functions are a very fast way to clean up data due to special
> expectation but think about the border of dilated/eroded areas - you
> shift the border of the eroded area and so you can lost information you
> are interested in.

Not necessarily true. You normally do an erosion (or several erosions) followed by the same number of dilations with the same structuring elements. That will remove small areas and smooth boundaries of remaining areas without shifting them too much.

Typical structuring elements for 2D images:

s = replicate(1,3,3)

or the cross

```
  +
 +++
  +
```

(3x3 array with indicated elements equal to one). You should experiment a bit. Standard textbooks on image processing will have a section on morphological image processing.

An alternative approach to reject 'noise ice' which might be characterised by having a very small area is to delete all 'connected' areas that are smaller than a given threshold. See the example of the 'label_region' function on how to do something like that.

Christian