bclose

keys:	**[from]**	*<number>*	source picture
	[to]	*<number>*	destination picture
	with	*<number>*	picture specifying structuring element data
	mask	*<number>*	mask picture to restrict processing to specified region
	times	*<number>*	number of times to apply structuring element data
	edge	*<number>*	edge value
options:	**source/output**		fix value of source/output edge pixels according to the value of the **edge** key

You use **bclose** to carry out generalised binary closing of foreground regions. The way in which the closing is carried out is determined by the structuring element specified by means of the **with** key. The output picture will contain the binary, closed result, which represents the complement of the area swept out by the reflection of the structuring element about the origin, while it is contained entirely within the background regions of the source image. Closing smoothes the boundary of foreground regions by filling in small concavities, it fills in narrow gaps between regions and it removes holes that are smaller than the structuring element. Closing is idempotent – closing an already closed image does not change the image.

Bclose has exactly the same keys and options as the commands **bdilate** and **berode** and it produces exactly the same result as using the command **bdilate** followed by **berode** with the same settings for the controlling keys and options, but in less time.

Examples

```
create 99 size 3 value 1; bclose 1 with 99
```

This example closes picture 1 with square, 3 by 3 structuring element.

```
bclose 1 2 with 99 times 3
```

This command is equivalent to the following.

```
bdilate 1 2 with 99 times 3; berode 2 with 99 times 3
```

bclose

Description

Closing a binary image with a structuring element is equivalent to dilating and then eroding the image with the same structuring element. For example,

```
0 0 0 0 0 0 0 0 0 0                                   0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0                                   0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 1 1 0 0                                     0 0 0 1 1 1 1 0 0
0 0 1 0 1 1 1 0 0           1 1 1                     0 0 1 1 1 1 1 0 0
0 0 1 1[1]1 1 0 0   (.)     1[1]1   =                 0 0 1 1[1]1 1 0 0
0 0 1 0 0 0 0 0 0           1 1 1                     0 0 1 1 1 1 1 0 0
0 0 0 0 1 1 1 0 0                                     0 0 0 0 1 1 1 0 0
0 0 0 0 0 1 0 0 0                                     0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0                                     0 0 0 0 0 0 0 0 0
```

where (.) denotes closing and [ ] marks the origin of the picture and the structuring element. Compare this with the result obtained by erosion followed by dilation,

```
0 0 0 0 0 0 0 0 0 0                                   0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0                                   0 0 1 1 1 1 1 1 0
0 0 0 1 0 1 1 0 0                                     0 1 1 1 1 1 1 1 0
0 0 1 0 1 1 1 0 0           1 1 1                     0 1 1 1 1 1 1 1 0
0 0 1 1[1]1 1 0 0   (+)     1[1]1   =                 0 1 1 1[1]1 1 1 0
0 0 1 0 0 0 0 0 0           1 1 1                     0 1 1 1 1 1 1 1 0
0 0 0 0 1 1 1 0 0                                     0 1 1 1 1 1 1 1 0
0 0 0 0 0 1 0 0 0                                     0 0 0 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0                                     0 0 0 0 1 1 1 0 0

0 0 0 0 0 0 0 0 0 0                                   0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 0                                     0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 0                                     0 0 0 1 1 1 1 0 0
0 1 1 1 1 1 1 1 0           1 1 1                     0 0 1 1 1 1 1 0 0
0 1 1 1[1]1 1 1 0   (-)     1[1]1   =                 0 0 1 1[1]1 1 0 0
0 1 1 1 1 1 1 1 0           1 1 1                     0 0 1 1 1 1 1 0 0
0 1 1 1 1 1 1 1 0                                     0 0 0 0 1 1 1 0 0
0 0 0 1 1 1 1 1 0                                     0 0 0 0 0 1 0 0 0
0 0 0 0 1 1 1 0 0                                     0 0 0 0 0 0 0 0 0
```

where (–) denotes erosion, (+) denotes dilation.

bclose

The source picture is treated as a binary image: zero values denote background pixels and non-zero values denote foreground pixels.

Each layer of the picture specified by means of the **with** key defines a structuring element to be applied to the source picture. Non-zero pixels define the components of the structuring element. Any zero pixels are ignored. The origin of the picture defines the origin of the structuring element.

For example,

```
0 0 0 0 0                    0 0 1 0 0              1
0 0 1 0 0          1         0 0 1 0 0              1
0 1[2]1 0    => 1[1]1        0 0 1 0 0      =>     [1]
0 0 1 0 0          1         0 0 1 0 0              1
0 0 0 0 0                    0 0 1 0 0              1
```

If there is more than one layer, the structuring elements thus defined are applied in sequence, starting with layer 1. This allows you to specify large structuring elements in decomposed form. You can also use the **times** key to apply the sequence of structuring elements more than once. The structuring element represented by a decomposed set is obtained by combining the decomposed set using dilation.

For example,

```
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1          1 1 1      1 1 1      1 1 1      1 1 1
1 1 1 1[1]1 1 1 1    =     1[1]1 (+)  1[1]1 (+)  1[1]1 (+)  1[1]1
1 1 1 1 1 1 1 1 1          1 1 1      1 1 1      1 1 1      1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1

1 1 1                          1
1[1]1        =     1[1]1 (+)   [1]
1 1 1                          1
```

As you can see, the 9 by 9 structuring element can be decomposed into four 3 by 1 structuring elements and four 1 by 3 structuring elements. As dilation is a commutative operation, the structuring elements can be applied in any order. The simplest way to apply this example is to create a two-layer picture with the 3 by 1 structuring element in one layer and the 1 by 3 structuring element in the other layer, specify this picture by means of the **with** key and set the **times** key to 4.

bclose

The **mask** key can be used to restrict processing to specified regions of the image. The options **source** and **output** and the **edge** key allow you to control edge effects. For more detailed information about these and related topics, consult the documentation for the commands **berode** and **bdilate**.

Notes

see also: **berode, bdilate**

Defaults and Ranges

keys/options	defaults	range
[from]	current picture, held in the variable *select*	valid picture number
[to]	source picture	valid picture number
with	none	valid picture number
mask	none	valid picture number
times	1	postive integer
edge	0	real number
source/output	do not process undefined pixels	