erode

keys:	**[from]**	*<number>*	source picture
	[to]	*<number>*	output picture
	times	*<number>*	force repeat count for erosion process
	neighbours	*<number>*	erode points with specified minimum number of clear neighbours
	with	*<number>*	picture containing user-supplied map functions
options:	**skeletonise**		erode object down to 8-connected skeleton
	ends		erode ends of protruding branches/hairs
	nodes		erode intersections from skeleton, separating branches
	outline		erode interior of object, leaving (8-connected) outline
	ol4		erode interior of objects, leaving 4-connected outline

Use **erode** to process a binary picture by removing a one-pixel border from an object, made up of non-zero pixels. This erosion process sometimes helps to clarify the relationship between objects or particles. Note that zero-value pixels are referred to as *clear* or *background* pixels and non-zero value pixels as *set* or *object* pixels.

Examples

```
erode display
```

This command strips a one-pixel wide border from all objects (by clearing pixels which have a clear neighbour).

```
erode 50 to 51 times 3
```

This command strips a three-pixel border from all objects (that is, it repeats the **erode** command three times).

```
erode neighbours 5
```

This command strips off border pixels which have at least five clear neighbours. This removes isolated points and lines without affecting the edge of extended objects.

erode

```
erode neighbours 8
```

This command clears isolated set pixels only.

```
erode skeleton
```

This command thins down objects to (8-connected) skeleton lines or curves.

```
erode ends
```

This command strips away all free ends (branches, hairs) from a skeleton or extended object.

```
erode display with 21
```

This command applies a user-defined operation defined in picture 21.

Description

The **erode** command is part of the *morphology* group of commands that perform morphological (shape) operations on binary pictures. Other morphology commands include **analyse**, **dilate** and **median**. You use **erode** on binary pictures, that is, pictures that have only two classes of value:

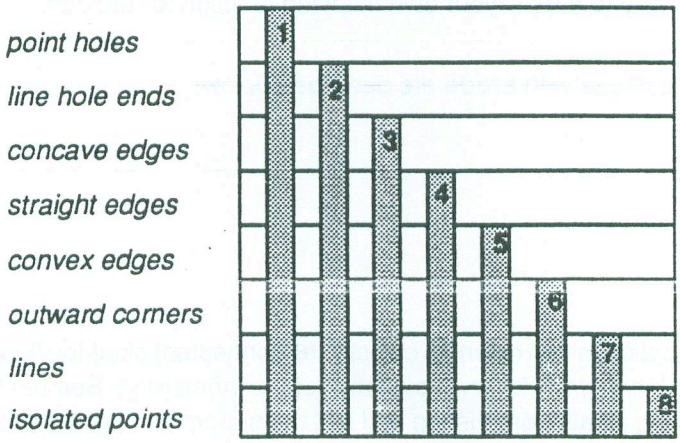
- zero for background pixels
- non-zero pixels belonging to the shape (object) of interest

Note that as Semper does not actually store pixels in single bit form, **erode** distinguishes between zero (*clear*) and non-zero (*set*) pixels instead. You can translate a picture into binary form using the **calculate** command, for example, **calculate :51>20** or **calculate :51<thr**.

By default, **erode** clears pixels that have at least one clear neighbour. You can use the **times** key to specify the number of times the **erode** command is repeated and so determine the depth of the erosion. If you set **times** to zero, **erode** repeats itself until the picture stabilizes (or until you *abandon* the operation). This is faster than putting **erode** in a **for** loop, as the command is able to omit picture rows that have already stabilized in previous passes. Note that the sequence **erode; dilate** is a useful way of smoothing object borders.

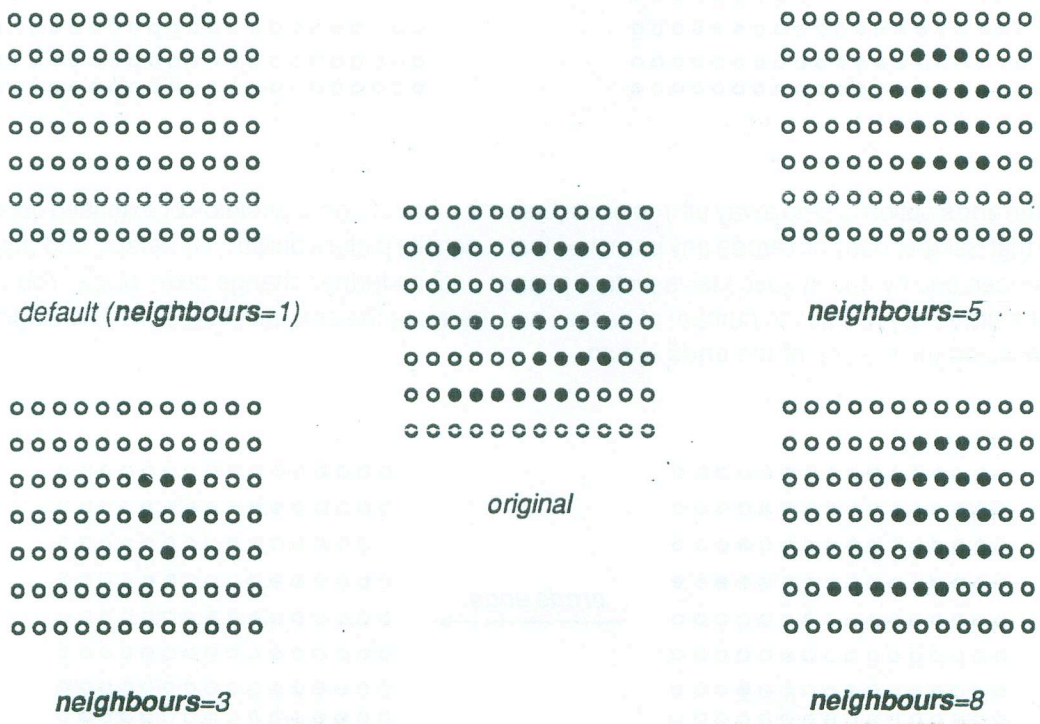
By default, **erode** clears pixels with at least one clear neighbour. You can use the **neighbours** key to perform more selective erosion as you increase this number towards the maximum of 8. The diagram given opposite shows the erosion of different features of an object according to the value of the **neighbours** key.

erode



Erosion process according to value of ***neighbours*** *key*

The following diagram shows the result of giving different values to the **neighbours** key for the same object configuration:



erode

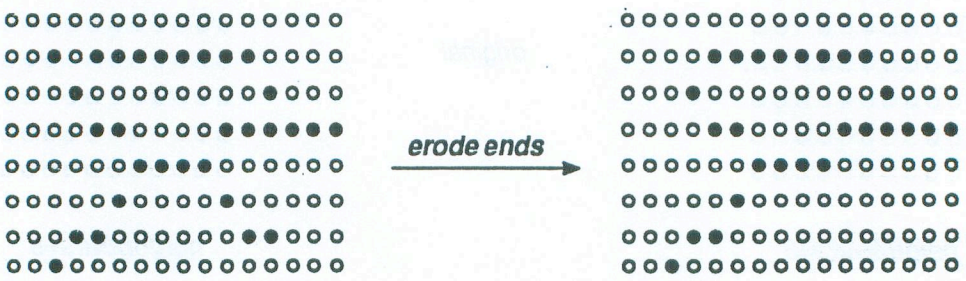
You can use the **with** key to supply your own mapping function for erosion.

The options that you can use with **erode** are described below:

- **skeletonise**
- **ends**
- **nodes**
- **outline, ol4**

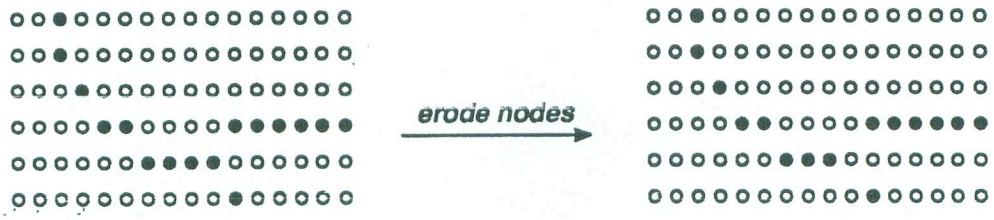
Use the **skeletonise** option to thin down an object to (8-connected) skeleton lines or curves. (Refer to *Appendix G: Pixel Connectivity* for an explanation of 8-connectivity). Semper thins down objects in four passes, removing pixels from the top and left, the bottom and right, the bottom and left and the top and right in turn, without affecting the 8-connectivity of objects. By default, Semper repeats this sequence until the picture stabilises but you can limit the number of sequences using the **times** key. The diagram below illustrates the effect of the **skeleton** option.

Use the **ends** option to strip away all free ends (branches, hairs) from a skeleton or extended object. Note that Semper does not erode any branches that reach the picture border. By default, end pixels are eroded one by one in successive picture passes until no further change takes place. You can use the **times** key to limit the number of passes and therefore the amount of erosion. The diagram below illustrates the use of the **ends** option.



erode

Use the **nodes** option to delete pixels from a skeleton at intersections, and so separate branches in a picture so that you can measure them separately, for example. The diagram below illustrates the effect of the **nodes** option.



Use the **outline** option to erode interior points of objects (in a single pass) leaving minimal 8-connected outlines. Use the **ol4** option to leave outlines that are 4-connected rather than 8-connected.

Notes

see also: **analyse, calculate, dilate, median**

Defaults and Ranges

keys/options	defaults	range
[from]	current picture, held in the variable *select*	valid picture number
[to]	source picture	valid picture number
times	1	zero or positive integer
neighbours	1 pixel	integer in range 1 to 8
with	*none*	valid picture number