

extract

keys:	[from]	<number>	source picture
	[to]	<number>	output picture
	size	<x>, <y>	dimensions of extracted subregion
	position	<x>, <y>	position/offset of subregion
	sampling	<number>	sampling interval within subregion
	angle	<number>	rotate a subregion through the specified angle in an anticlockwise direction
	with	<number>	<i>Plist</i> picture listing source positions at which output samples are required
	mark	<number> <yes or no>	mark source subregion, if <i>with</i> , mark sample points
	mkmode	<number>	if <i>with</i> , mark mode
	mksize	<number>	if <i>with</i> , mark size
options:	left/right, top/bottom		subregion position
	uv		extract samples on a lattice defined by the variables <i>u</i> , <i>u2</i> and <i>v</i> , <i>v2</i>
	average		average source samples over undersampled regions
	verify		print information about the process at the console
	nneighbour/bicubic		use nearest neighbour extraction or bicubic interpolation instead of bilinear interpolation

Use **extract** to extract rotated, (de)magnified or skewed regions of a picture. You can also use **extract** to rotate pictures, to repeat pictures in periodic arrays, and to average pixels in small blocks while undersampling.

Examples

```
extract size 100 top left to 2
```

This command extracts a 100 square region top left to picture 2 (like **cut**).

extract

```
extract size 512 display
```

This command repeats a small current picture periodically on the display.

```
xwires curve sampling 1; extract 1 2 with 999 nneighbour
```

This command evenly extracts spaced samples from picture 1, along an arbitrary curve drawn with the cursor, using nearest neighbour extraction.

```
extract 1 2 size 128 sampling 2 average
```

This command averages (the central 256 square of) picture 1 in 2 by 2 blocks to form a 128 square picture 2.

```
extract to display size 100 sampling .75 position x,y
```

This command interpolates a region 75 points square centred at x,y to form a 100 square display.

```
extract to 50 size 300,1 angle pi/4 position x,y; display
```

This command displays a diagonal line scan through x,y.

```
xwires curve sampling 1; extract 1 2 with 999
```

This command extracts evenly spaced samples from picture 1, along an arbitrary curve drawn with the cursor.

```
u=u/64,u2/64 v=v/64,v2/64; extract size 64 uv
```

This command extracts a single unit cell of a periodic image with lattice base vectors u, v .

Description

Use the **extract** command to perform the following basic actions:

- extract a region (using bilinear interpolation if necessary)
- expand the extracted region by **sampling**
- rotate the region using **angle**

Note that if the specified region overflows the source picture boundary, the source picture is treated as continuing periodically in both directions.

extract

Use the **with** key to perform an arbitrary geometric transformation. Supply a *Plist* picture number for **with** and Semper generates an output picture with the same column and row length as the *Plist*, but with each position replaced by a value interpolated from the source picture at that position. For example, the following sequence of commands resamples picture 50 on a radial polar grid (radius 0–49, angle 0–359 degrees).

```
create 3 plist size 360,50,2
origin bottom left
calculate ifelse(z,y*sin(rad(x)),y*cos(rad(x)))
extract 50 to 51 with 3
```

See *Appendix A: Picture Types* for details of *Plist* pictures.

Use the **mark** key to mark the source subregion or, if you specify **with**, the sample points. Use the keys **mkmode** and **msize** to specify the style of marking. Refer to *Appendix C, Semper Keys and Options* for details of the keys **mark**, **mkmode** and **mksize**.

The **uv** option causes **extract** to distort the extracted region so that the vector $(u,u2)$ defined by the current value of the variables becomes one point to the right in the output and $(v,v2)$ one point upwards. **sampling** and **angle** are ignored in these circumstances.

Apply the **average** option only when undersampling without interpolation, as it may be useful for preserving the original signal-to-noise ratio. The averaged blocks, exceptionally, have the top left rather than centre at the nominal sampling position, which allows you to reduce a $2n$ square picture to a n square without an additional **position** adjustment.

The **extract** command uses bilinear interpolation between the four neighbouring source pixels to produce the output values at each sample position but you can also use the **nneighbour** option to specify nearest neighbour extraction or the **bicubic** option to specify bicubic interpolation:

- If you specify the **nneighbour** option, the value of the nearest neighbouring source pixel is taken as the output value.
- If you specify the **bicubic** option, the output pixel values are generated using bicubic interpolation between the 4 by 4 array of source pixels that surrounds each sample position.

Roughly speaking, nearest neighbour extraction is twice as fast as bilinear interpolation and bicubic interpolation is twice as slow. Nearest neighbour extraction is useful when it is necessary to avoid the artefacts caused by the interpolation process. Bicubic interpolation avoids the sharp discontinuities of intensity gradient which becomes more apparent when using bilinear interpolation in conjunction with very small sampling intervals (high magnification).

Semper 6 Command Reference

extract

Note that you may find **extract** slow in operation for large pictures, if the rows to be extracted lie at large angles to those of the source. In this case, use **rotate** to perform large angle rotations (even if initially you have to extract a larger region than you want).

If the source picture origin is included in the extracted region, it is recorded as the origin of the output (rounded to the nearest pixel), unless **extract with** is used.

Notes

display marking:	extracted region
multi-layer pictures:	layers processed independently
forms used internally:	integer, fp, complex
variables used:	$u, u2, v, v2$ (if uv , base vectors defining the size and shape of a skewed region)
see also:	cut, rotate, warp

Defaults and Ranges

keys/options	defaults	range
[from]	current picture, held in the variable <i>select</i>	valid picture number
[to]	source picture	valid picture number
size	whole picture	less than or equal to the size of the picture (integers)
position	position 0,0	real numbers
sampling	1 sampling interval	positive real number
angle	angle 0	real number in range 0 to 2π
mark	mark off	see <i>Appendix C</i>
mkmode	1 (upright cross)	integer in range 1 to 5
mksize	2	positive integer
verify	verification off	
nneighbour/bicubic	bilinear	