xcf

keys:	**[from]**	*<number>*	first picture to be cross-correlated
	[to]	*<number>*	output picture
	with	*<number>*	second picture to be cross-correlated
	via	*<number>*	intermediate picture, receiving Fourier transform of ***with*** picture
	radius	*<number>*	if ***search***, radius around peak over which a local centre-of-mass is calculated to replace the raw peak position
	mark	*<number>* *<yes> or <no>*	mark cross-correlation peaks on display or, if ***radius***, the circular region
	mkmode	*<number>*	mark mode
	mksize	*<number>*	mark size
options:	**search**		search *xcf* for peak, returning position as *x*, y
	lowest		if ***search***, locate lowest rather than highest point
	negative		if ***search*** and ***radius***, include negative pixels in centre-of-mass search
	squared		if ***search*** and ***radius***, find centre-of-mass of squared pixels
	iterated		if ***search*** and ***radius***, repeat the centre-of-mass location once, with a region shifted to be centred at the centre-of-mass found initially
	verify		verify cross-correlation process at the console

Use **xcf** to calculate cross-correlation functions between pairs of pictures. A cross-correlation function allows you to:

- find a lateral shift needed to align (register), the pictures
- locate motifs occurring at many places in a picture
- detect features buried in high noise levels
- measure signal-to-noise ratios

xcf creates an output picture of class *Correlation*, in floating-point form.

xcf

Examples

```
xcf 1 with 2 to 3; extract 2 to 3 @xy
```

This command calculates the *xcf* of pictures 1 and 2 in picture 3 (class *Correlation*), and produces a shifted version of picture 2 in register with picture 1.

```
xcf 1 with 2; type x,y,t,root(t/(1-t))
```

This command replaces picture 1 with the *xcf* between pictures 1 and 2, and reports the correlation peak position and height, and the signal-to-noise ratio, on the assumption that they are identical apart from noise.

```
fourier 1; xcf 2 with 1; cut 2 size 400 @xy
xcf 3 with 1; cut 3 size 400 @xy
xcf 4 with 1; cut 4 size 400 @xy
```

This sequence of commands produces subregions from pictures 2, 3 and 4 in turn, that are in register with picture 1.

```
create 1 size 512; paste 2; mask radius 35; xcf with 3 nosearch
```

This command produces a correlation map showing positions in a 512 square picture 3 that matches a radius 35 motif in picture 2.

Description

The cross-correlation (*xcf*) of picture *f* (**from**) with picture *w* (**with**) is a map of the **with** picture showing how well *f* matches *w* when its centre is shifted to the position being mapped. The output picture may indicate the following:

- if *f* and *w* are similar apart from a relative displacement, the output picture contains a single strong peak. The position of the peak identifies the displacement.
- if *w* contains many copies of a motif at the centre of *f*, the *xcf* has local peaks marking the position of each such copy.

xcf also searches for the highest correlation peak (unless you specify **nosearch**). You can also search for the lowest peak using the **lowest** option. If you specify a value for the **radius** key, **xcf** also finds a local centre-of-mass around the peak.

If you use the **mark** key to specify the display, **xcf** marks the cross-correlation peak on the display, in the style and size determined by **mkmode** and **mksize**. For details of the keys **mark**, **mkmode** and **mksize** refer to *Appendix C, Semper Keys and Options*.

xcf

In more detail, the form of the normalized *xcf* calculated by **xcf f with w** is:

$$xcf(x') = \frac{\langle f(x)w(x-x')\rangle - \langle f\rangle\langle w\rangle}{\sqrt{(\langle f^2\rangle - \langle f\rangle^2)(\langle w^2\rangle - \langle w\rangle^2)}}$$

where <..> denotes an average over the whole picture. The normalization makes the result independent of the grey-scaling of the two pictures (unchanged by constant addition, subtraction, multiplication or division). The results lie in the range –1 to 1:

- values near zero indicate there is little agreement between the pictures
- values near 1 indicate that the pictures agree closely
- values near –1 indicate that the pictures agree closely but with opposite contrast

In general, correlation functions are remarkably powerful at detecting a match in the presence of noise. (Correlating images *n* points square detects a common signal buried in noise with a level *root(n)/2* times that of the signal). Note that substantial differences in imaging conditions between *f* and *w* do not usually prevent **xcf** from finding a match.

Note that the options **negative**, **squared** and **iterated** are relevant only in the centre-of-mass mode, that is, when you set **radius**.

Further Information

To calculate the cross average, Semper transforms both source pictures:

- **from** to the output picture **to**
- **with** to the intermediate picture **via**

Semper multiplies the conjugate of the first transform with the second transform before inverse transforming the product.

You can also supply pictures that are already transformed, that is *Fourier* pictures instead of *Images* and **xcf** picks up the calculation at the appropriate point. This allows you to include additional filters to manipulate *xcf* peak shapes and/or noise levels and is also useful if you are aligning several pictures in turn with a reference picture, since you need only transform the reference once.

If the images are not correctly registered, correlation levels are reduced in proportion to a fraction of the image area that is common to both images. If this matters, for example, because you are using the peak height to deduce the signal-to-noise ratio as in the second command example, you can correct the height before proceeding by a sequence of commands such as:

```
xcf..; pcb; t = t * ncols/(ncols-mod(x)) * nrows/(nrows-mod(y))
```

xcf

When you use discrete transform methods, as above, to calculate *xcfs*, the correlated *Images* and the generated *Correlation* are both treated as if repeated periodically in both directions. This means that features shifted off one edge of an *Image* reappear at the opposite edge, and may give rise to spurious correlation. Also, if the *Images* are mis-registered by more than half the field of view, the *Correlation* peak (which should be displaced off one edge of the picture) in fact reappears at the opposite edge in a spurious position. To prevent this, align a larger area in the first instance, more coarsely sampled if necessary.

Notes

restrictions:	image sizes must be powers of two unsuitable for direct output to display
display marking:	peak position, region searched for centre-of-mass
multi-layer pictures:	faulted
forms used internally:	fp (complex in transforms)
variables set:	*x, y* (position of xcf peak found) *t* (height of xcf peak found – or mass if **radius**)
see also:	**fourier**, **pcb**

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
via	picture 999	valid picture number
radius	radius 0	positive real number
mark	mark off	see *Appendix C*
mkmode	1 (upright cross)	integer in the range 1 to 5
mksize	2	positive integer
search	search on	
negative	negative pixels included in search	
verify	verification off	