Interpolation for scaling, rotation, perspective and morphing

The distance between two source pixels is 1. Using scalefactor S=8 means: 7 pixels must be inserted in each direction. This is done by nearest neighbour interpolation, bilinear, biquadratic or bicubic interpolation. Using the zoom shows that indeed one left box-like pixel is substituted by 8x8 = 64 pixels in the right image.



Interpolation principles (illustration on the next page)

The green plane represents 4x4 pixels of an original RGB image. The image itself is not illustrated.

The vertical stakes show the values of one of the three RGB colors, e.g. green.

The stakes are covered by a smooth surface like an elastic skin.

For biquadratic or bicubic Interpolation the surface is parabolic of second or third order.

For bilinear Interpolation, the stakes are connected by twisted planes. The roof looks then polygonal instead of smooth. This is for many cases good enough, mainly if the interpolated image does not have considerably more pixels.

The image shall have a different number of pixels, but the geometrical size is the same. The green plane represents now 12x12 pixels for the image.

Intermediate pixels can be calculated by taking the values for e.g. green from the smooth

surface.

This method can be applied for scaling, rotation, perspective transformations and morphing. The example shows upsampling by factor S=3. Any factor is possible, e.g. S=2.345.

An accurate description of interpolation algorithms can be found here: http://docs-hoffmann.de/bicubic03042002.pdf

> Gernot Hoffmann November 11 / 2001- February 09 / 2013 Website Load browser / Click here

Upsampling. View by 72dpi and zoom 100% or 200%





Interpolation for downsampling

ZEBRA uses a linear interpolator. For downsampling this interpolator works as an averager. This is accurate for scalefactors down to about S=0.5. Downsampling for e.g. S=0.3 is done by downscaling first by S=0.6 and then in a second step by S=0.5. The averager is applied twice.





