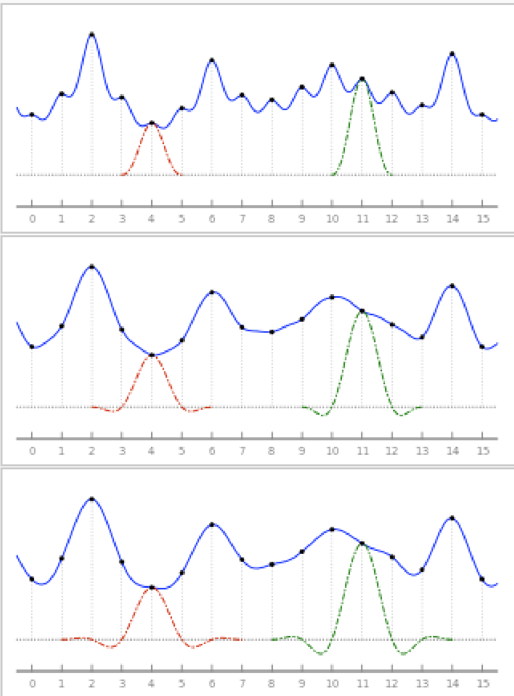
Lanczos resampling

From Wikipedia, the free encyclopedia

Lanczos resampling and **Lanczos filtering** are two applications of a mathematical formula. It can be used as a low-pass filter or used to smoothly interpolate the value of a digital signal between its samples. In the latter case it maps each sample of the given signal to a translated and scaled copy of the **Lanczos kernel**, which is a sinc function windowed by the central lobe of a second, longer, sinc function. The sum of these translated and scaled kernels is then evaluated at the desired points.



Partial plot of a discrete signal (black dots) and of its Lanczos interpolation (solid blue curve), with size parameter *a* equal to 1 (top), 2 (middle) and 3 (bottom). Also shown are two copies of the Lanczos kernel, shifted and scaled, corresponding to samples 4 and 11 (dashed curves).

Lanczos resampling is typically used to increase the sampling rate of a digital signal, or to shift it by a fraction of the sampling interval. It is often used also for multivariate interpolation, for example to resize or rotate a digital image. It has been considered the "best compromise" among several simple filters for this purpose.[1]

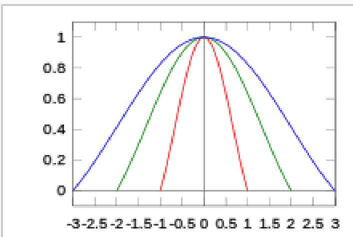
The filter is named after its inventor, Cornelius Lanczos (Hungarian pronunciation: [ˈlaːntsoʃ] or "lan-tsoh-sh").

Contents [hide]

Definition [edit]

Lanczos kernel [edit]

The effect of each input sample on the interpolated values is defined by the filter's reconstruction kernel $L(x)$, called the Lanczos kernel. It is the normalized sinc function $\operatorname{sinc}(x)$, windowed (multiplied) by the **Lanczos window,** or **sinc window**, which is the central lobe of a horizontally-stretched sinc function $\operatorname{sinc}(x/a)$ for $-a \le x \le a$.

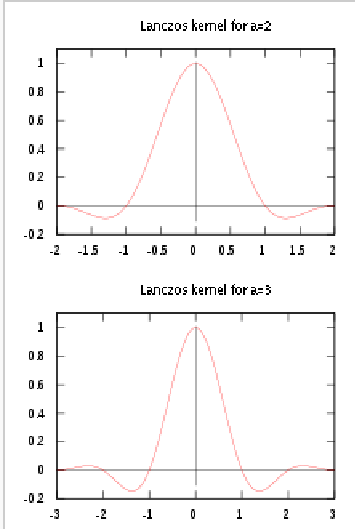


Lanczos windows for *a* = 1, 2, 3.

$$L(x) = \begin{cases} \operatorname{sinc}(x)\,\operatorname{sinc}(x/a) & \text{if } \; -a < x < a \\ 0 & \text{otherwise} \end{cases}$$

Equivalently,

$$L(x) = \begin{cases} 1 & \text{if } x = 0 \\ \dfrac{a\sin(\pi x)\sin(\pi x/a)}{\pi^2 x^2} & \text{if } 0 < |x| < a \\ 0 & \text{otherwise} \end{cases}$$



Lanczos kernels for the cases $a = 2$ and $a = 3$. Note that the function obtains negative values.

The parameter a is a positive integer, typically 2 or 3, which determines the size of the kernel. The Lanczos kernel has $2a-1$ lobes, a positive one at the center and $a-1$ alternating negative and positive lobes on each side.

Interpolation formula [edit]

Given a one-dimensional signal with samples s_i, for integer values of i, the value $S(x)$ interpolated at an arbitrary real argument x is obtained by the discrete convolution of those samples with the Lanczos kernel;[2] namely,

$$S(x) = \sum_{i=\lfloor x\rfloor - a + 1}^{\lfloor x\rfloor + a} s_i L(x-i),$$

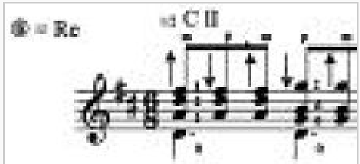
where a is the filter size parameter and $\lfloor x \rfloor$ is the floor function. The bounds of this sum are such that the kernel is zero outside of them.

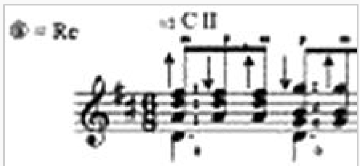
Properties [edit]

As long as the parameter a is a positive integer, the Lanczos kernel is continuous everywhere, and its derivative is defined and continuous everywhere (even at $x = \pm a$, where both sinc functions go to zero). Therefore, the reconstructed signal $S(x)$ too will be continuous, with continuous derivative.

The Lanczos kernel is zero at every integer argument x, except at $x = 0$, where it has value 1. Therefore, the reconstructed signal exactly interpolates the given samples: we will have $S(x) = s_i$ for every integer argument $x = i$.

Multidimensional interpolation [edit]


The *incipit* of a piece by Gaspar Sanz. Original, low quality expansion with JPEG artifacts. Open the picture to see the details.


The same image resampled to five times as many samples in each direction, using Lanczos resampling. Pixelation artifacts were removed changing the image's transfer function. Open the picture to see the details.

Lanczos filter's kernel in two dimensions is simply the product of two one-dimensional kernels:[2][*dubious – discuss*]

$$L(x,y) = L(x)\cdot L(y).$$

Given a two-dimensional signal s_{ij} defined at integer points (i,j) of the plane (e.g. intensities of pixels in a digital image), the reconstructed function is

$$S(x,y) = \sum_{i=\lfloor x\rfloor - a + 1}^{\lfloor x\rfloor + a} \sum_{j=\lfloor y\rfloor - a + 1}^{\lfloor y\rfloor + a} s_{ij} L(x-i) L(y-j).$$

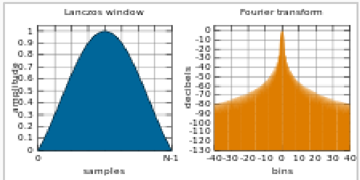
When resampling a two-dimensional signal at regularly spaced points (x,y), one can save some computation by resampling the entire signal along a single axis, then resampling the resulting two-dimensional signal along the other axis.

These formulas generalize to signals with three or more dimensions, in the obvious way.

Evaluation [edit]

Advantages [edit]

The theoretically optimal reconstruction filter for band-limited signals is the sinc filter, which has infinite support. The Lanczos filter is one of many practical (finitely supported) approximations of the sinc filter. Each interpolated value is the weighted sum of $2a$ consecutive input samples. Thus, by varying the $2a$ parameter one may trade computation speed for improved frequency response. The parameter also allows one to choose between a smoother interpolation or a preservation of sharp transients in the data. For image processing, the trade-off is between the reduction of aliasing artefacts and the preservation of sharp edges. Also as with any such processing, there are no results for the borders of the image. Increasing the length of the kernel increases the cropping of the edges of the image.



A discrete Lanczos window and its frequency response; see Window function for comparison with other windows.

The Lanczos filter has been compared with other interpolation methods for discrete signals, particularly other windowed versions of the sinc filter. Turkowski and Gabriel claimed that the Lanczos filter (with a = 2) the "best compromise in terms of reduction of aliasing, sharpness, and minimal ringing", compared with truncated sinc and the Bartlett, cosine-, and Hann-windowed sinc, for decimation and interpolation of 2-dimensional image data.[1] According to Jim Blinn, the Lanczos kernel (with a = 3) "keeps low frequencies and rejects high frequencies better than any (achievable) filter we've seen so far."[3]

Lanczos interpolation is a popular filter for "upscaling" videos in various media utilities, such as AviSynth[4] and FFmpeg.[5]

Limitations [edit]

Since the kernel assumes negative values for $a > 1$, the interpolated signal can be negative even if all samples are positive. More generally, the range of values of the interpolated signal may be wider than the range spanned by the discrete sample values. In particular, there may be ringing artifacts just before and after abrupt changes in the sample values, which may lead to clipping artifacts. However, these effects are reduced compared to the (non-windowed) sinc filter. For a=2 (a three lobed kernel) the ringing is <1%.

The method is one of the interpolation options available in the free software: Gnu Image Manipulation Program (GIMP). One way to visualise the ringing effect is to rescale a black and white block graphic and select Lanczos interpolation.

When using the Lanczos filter for image resampling, the ringing effect will create light and dark halos along any strong edges. While these bands may be visually annoying, they help increase the perceived sharpness, and therefore provide a form of edge enhancement. This may improve the subjective quality of the image, given the special role of edge sharpness in vision.[6]

In some applications, the low-end clipping artifacts can be ameliorated by transforming the data to a logarithmic domain prior to filtering. In this case the interpolated values will be a weighted geometric mean, rather than an arithmetic mean, of the input samples.

The Lanczos kernel does not have the partition of unity property. That is, the sum $U(x) = \sum_{i \in \mathbb{Z}} L(x - i)$ of all integer-translated copies of the kernel is not always 1. Therefore, the Lanczos interpolation of a discrete signal with constant samples does not yield a constant function. This defect is most evident when a = 1. Also, for a = 1 the interpolated signal has zero derivative at every integer argument. This is rather academic, since using a single lobe kernel (a=1) loses all the benefits of the Lanczos approach and provides a poor filter. There are many better single-lobe, bell-shaped windowing functions.

See also [edit]

- Bicubic interpolation
- Bilinear interpolation
- Spline interpolation
- Nearest-neighbor interpolation
- Sinc filter

References [edit]

1. ^ a b Ken Turkowski and Steve Gabriel (1990). "Filters for Common Resampling Tasks". In Andrew S. Glassner. *Graphics Gems I*. Academic Press. pp. 147–165. ISBN 978-0-12-286165-9. CiteSeerX: 10.1.1.116.7898.
2. ^ a b Wilhelm Burger, Mark J. Burge (2009). *Principles of digital image processing: core algorithms*. Springer. pp. 231–232. ISBN 978-1-84800-194-7.
3. ^ *Jim Blinn's corner: dirty pixels*. Morgan Kaufmann. 1998. pp. 26–27. ISBN 978-1-55860-455-1.
4. ^ "Resize". Avisynth. 2015-01-01. Retrieved 2015-07-27.
5. ^ "A How To guide: Upconverting video using FFDShow - Neowin Forums". Neowin.net. 2006-04-18. Retrieved 2012-07-31.
6. ^ "IPOL: Linear Methods for Image Interpolation". Ipol.im. 2011-09-27. Retrieved 2012-07-31.

External links [edit]

- Anti-Grain Geometry examples: `image_filters.cpp` shows comparisons of repeatedly resampling an image with various kernels.
- imageresampler: A public domain image resampling class in C++ with support for several windowed Lanczos filter kernels.

Categories: Signal processing | Multivariate interpolation