L. Yaroslavsky

FAST SIGNAL SINC-INTERPOLATION AND ITS APPLICATIONS IN IMAGE PROCESSING

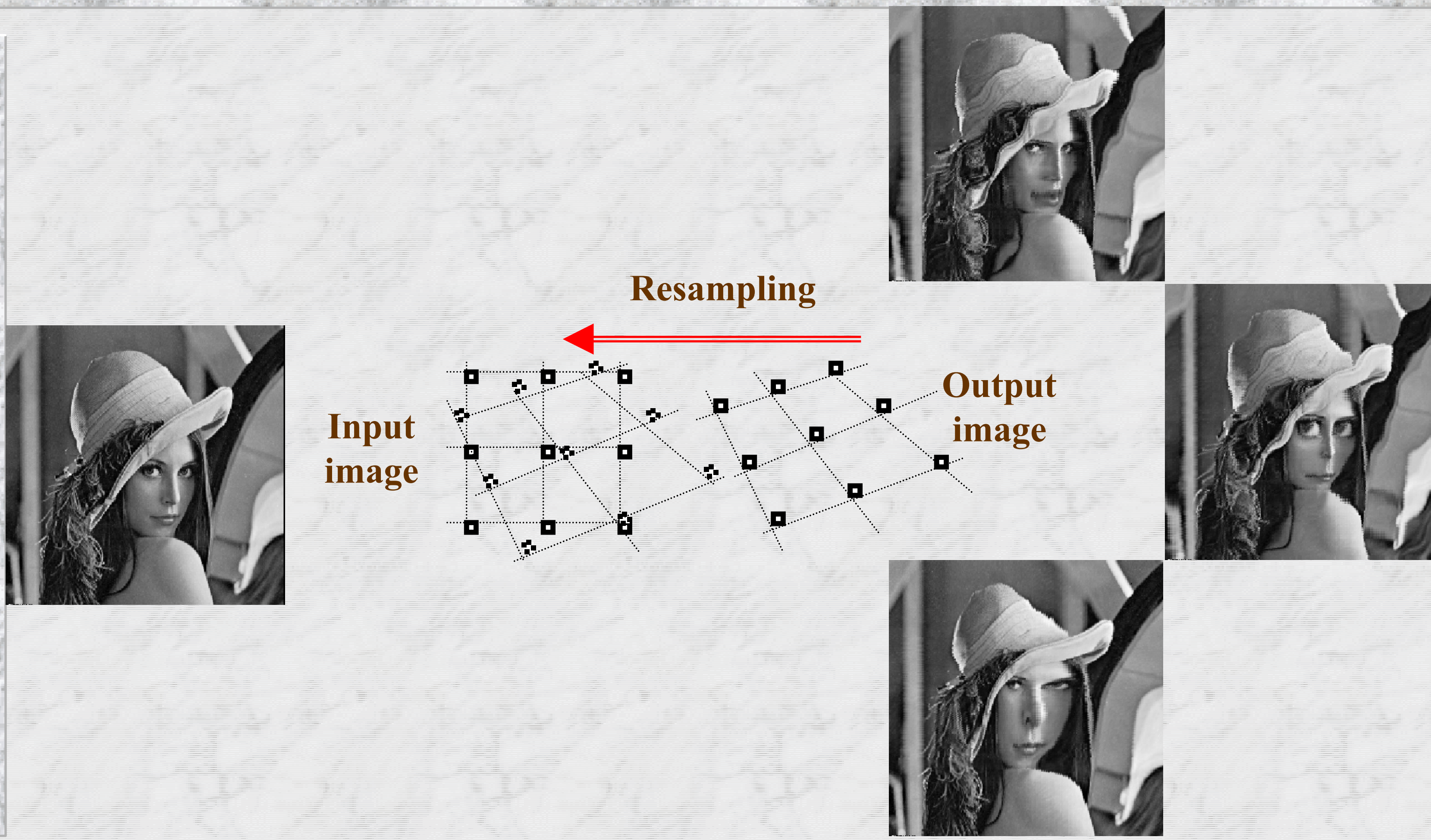
OUTLINE

- Topicality and principles of signal/image resampling
- Sinc-interpolation as a gold standard for convolution based interpolation
- Fast SDFT based sinc-inerpolation
- Examples of applications
- Sinc-interpolation in DCT domain
- Sinc-interpolation in sliding window

Signal/Image Resampling Applications

- Signal/image sampling standard conversion
- Signal “fractional” delay
- Image co-ordinate conversion; nonuniform-to-uniform raster resampling
- Signal/image localization and alignment with sub-pixel accuracy
- Radon Transform and tomographic reconstruction
- 3D imaging

Image resampling: basic principle



Signal interpolation computational methods

- **Nearest neighbor (zero order) interpolation:**

$$\vec{a}_{int}^{(0)} = conv\left[kron\left(\vec{a},\delta_L\right), ones(L)\right]; \qquad \delta(L) = [1,0,\underbrace{0,..,0}_{L}]$$

- **Linear (bilinear) interpolation:**

$$\vec{a}_{int}^{(1)} = conv\left[conv\left[kron\left(\vec{a},\delta_L\right), ones(L)\right], ones(L)\right].$$

- ***R*-th order spline interpolation:**

$$\vec{a}_{int}^{(R)} = conv\left[conv\cdots\left[conv\left[kron\left(\vec{a},\delta_L\right), ones(L)\right], ones(L)\right]\right].$$

- **Sinc-interpolation:**

$$a(x) = \sum_{k=-\infty}^{\infty} \alpha_k \,\mathrm{sinc}\left[\pi\left(x - k\Delta x\right)/\Delta x\right]$$

TWO APPROACHES:
Convolution based and Fourier based

Sinc-interpolation versus spline interpolation: an opinion

For a long time, sinc interpolation has been the holy grail of geometrical operations. ... The sinc-function provides error-free interpolation of bandlimited functions. There are two difficulties associated with this statement. The first one is that the class of bandlimited functions represents but a tiny fraction of all possible functions; moreover, they often give a distorted view of the physical reality – think of transition air/matter in CT scan: ... this transition is abdrupt and cannot be expressed as a bandlimited function.... The second difficulty is that the support of the sinc-function is infinite. An infinite support is not too bothering, provided an efficient algorithm can be found to implement interpolation with another equivalent basis function that has a finite support. This is exactly the trick we used with B-splines and o-Moms" (Ph. Thevenaz, Th. Blu, M. Unser, Interpolation Revisited, IEEE Trans. on Medical Imaging, V. 19, No. 7, July 2000)

The opinion that sinc-interpolation assumes band-limited functions is a fallacy. The meaning of the sampling theorem is this:

Given samples of a continuous function , there is no better approximation of the function with convolution bases functions than by a bandlimited function obtained by sinc-interpolation of its samples.

Sinc-interpolation: the gold standard for discrete linear interpolation of sampled data

Discrete Sampling Theorem:

Let a signal of *LN* samples is sampled with sampling interval of *L* samples. For $k_1 = 0,...,N-1$; $k_2 = 0,...,L-1$ the sampled signal can be written as $a_{k_1}\delta(k_2)$.

Compute its DFT:

$$a_{k_1}\delta(k_2) \xleftrightarrow{DFT} \frac{1}{\sqrt{LN}}\sum_{k_2}^{L-1}\sum_{k_1=0}^{N-1} a_{k_1}\delta(k_2)\exp\left[i2\pi\frac{(k_1 L + k_2)}{LN}r\right] =$$

$$\frac{1}{\sqrt{LN}}\sum_{k_1=0}^{N-1}\sum_{k_2=0}^{L-1} a_{k_1}\delta(k_2) exp\left[i2\pi\frac{(k_1 L + k_2)}{LN}r\right] =$$

$$\frac{1}{\sqrt{LN}}\sum_{k_1=0}^{N-1} a_{k_1}\, exp\left[i2\pi\frac{k_1}{N}r\right] = \frac{1}{\sqrt{L}}\alpha_{(r) mod\ N}$$

This means that sampling discrete signal results in periodical replication of its spectrum with the number of replicas equal to sampling interval.

Discrete sinc-interpolation: the gold standard for convolution based interpolation of sampled data

Signal interpolation by (periodical) digital convolution:

$$\tilde{a}_k = \sum_{k_1=0}^{N-1}\sum_{k_2=0}^{L-1} a_{k_1}\delta(k_2) h_{\text{int}}\left(k_{\text{mod}\,NL} - k_1 N + k_2\right)$$

In DFT domain:

$$DFT\left(\{\tilde{a}_k\}\right) = \tilde{\alpha}_r = \left(\alpha_{(r_1)\text{mod}\,K_1}\right)\cdot DFT\left(\{h_k\}\right)$$

Therefore, the only way to avoid aliazing and signal distortions is signal ideal low pass filtering. For N-odd number:

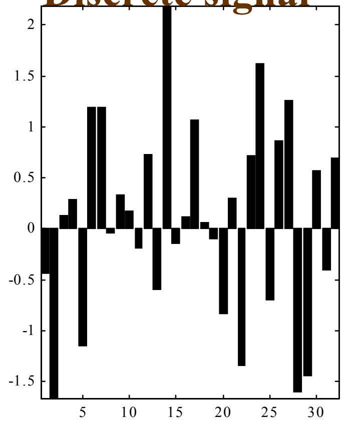
$$\left[1 - rect\frac{r - (N+1)/2}{LN - N - 1}\right]\alpha_{(r)\text{mod}\,N} \xleftrightarrow{IDFT} \frac{1}{\sqrt{L}}\sum_{k_1=0}^{N-1} a_n \mathrm{sincd}\left(\mathrm{N};\mathrm{N};(\mathrm{k} - \mathrm{k}_1 L)\right)$$

$$\mathrm{sincd}(\mathrm{K};\mathrm{N};\mathrm{x}) = \frac{\sin(\pi K x / N)}{N\sin(\pi x / N)}$$

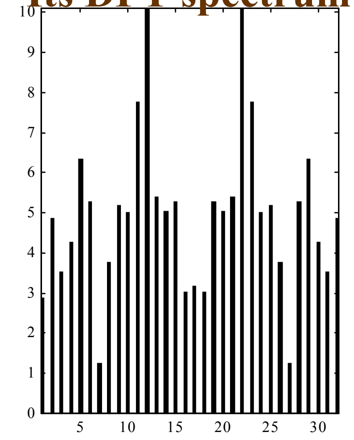
Problem: bounary effects due to the periodical convolution

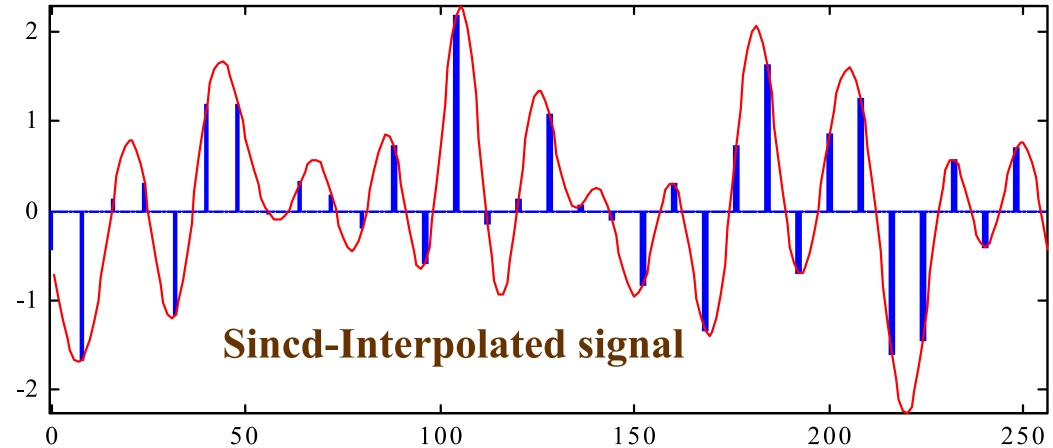
Discrete sinc-interpolation: a gold standard for interpolation of sampled data

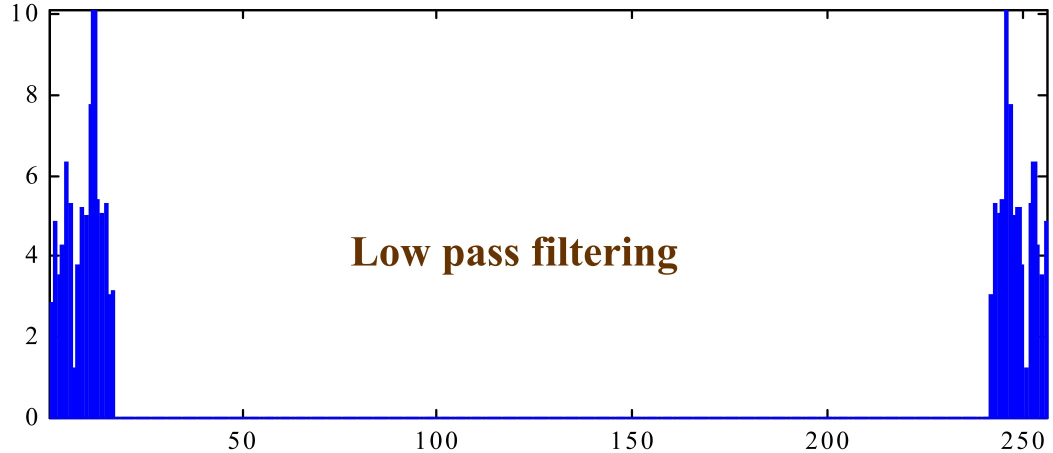
Discrete signal



Its DFT spectrum







Nearest neighbor, Bilinear, Bicubic spline and Sinc-image interpolation

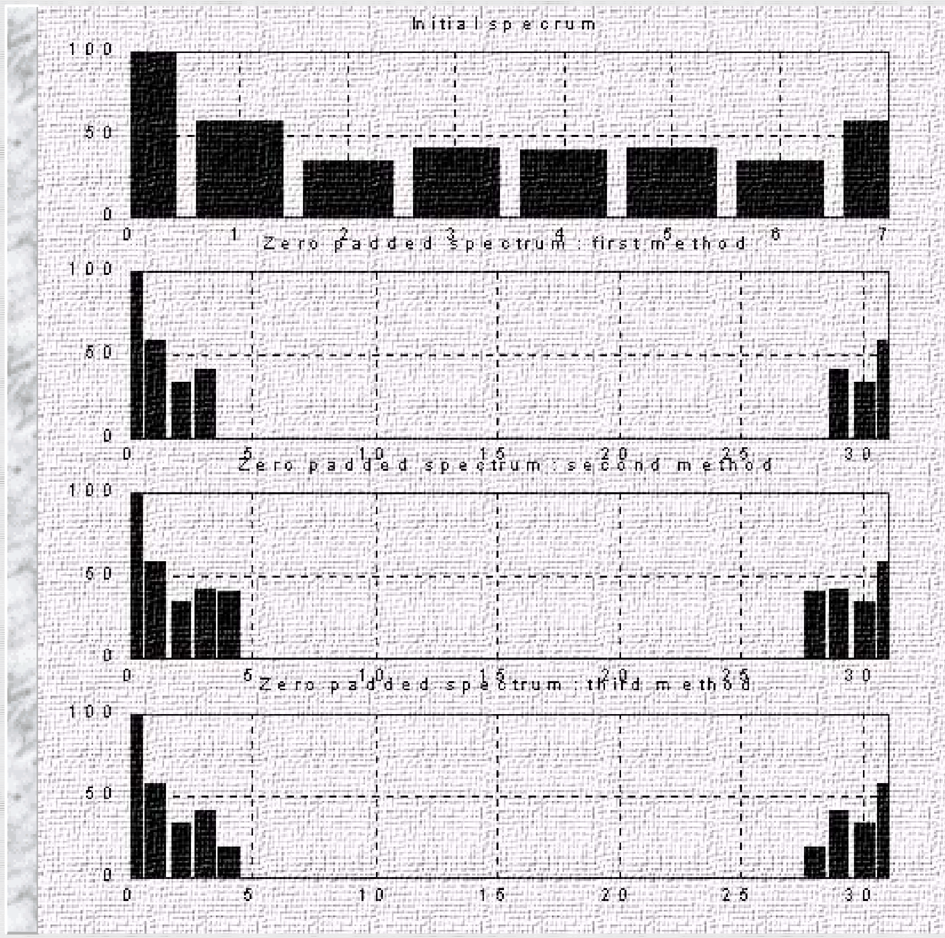
Nearest neighbor interpolation

Bilinear interpolation

Bicubic spline interpolation

Sinc-interpolation

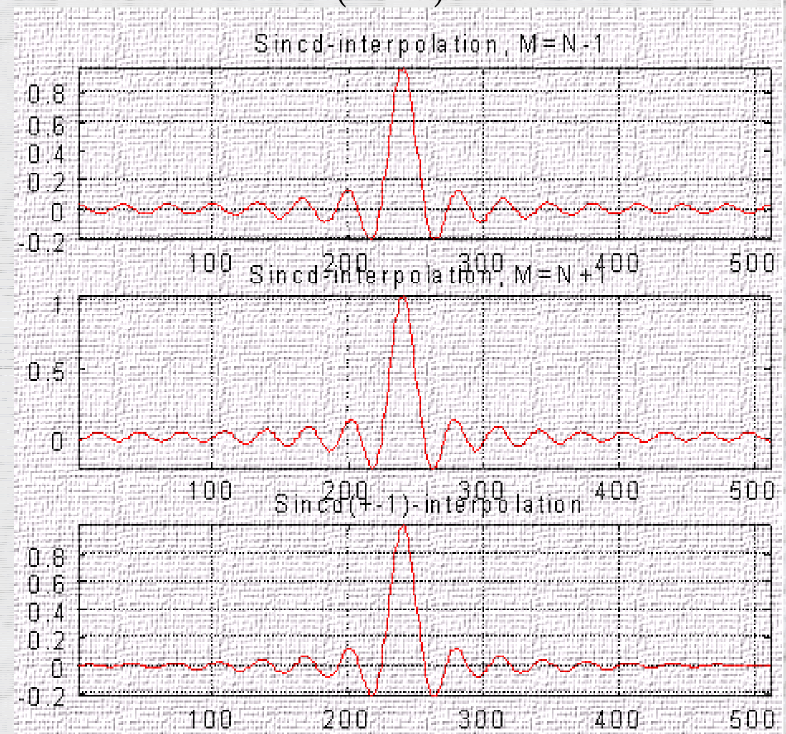
Computationl algorithms for disctere sinc-interpolation: Zero Padding Method



Interpolation functions:

$$\tilde{a}_k = \frac{1}{\sqrt{L}} \sum_{k_1=0}^{N-1} a_{k_1}\, sincd\left(M; N; \left(k - k_1 L\right)\right);$$

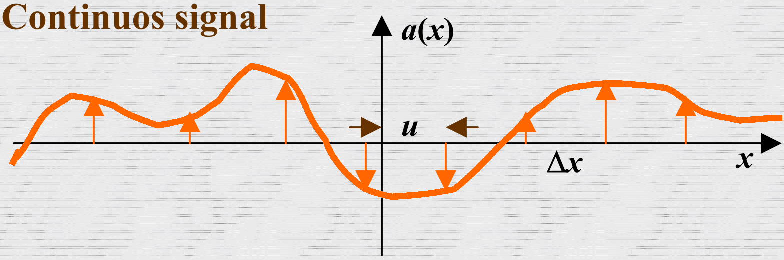
$$sincd\left(K; N; x\right) = \frac{\sin\left(\pi K x / N\right)}{N \sin\left(\pi x / N\right)};\ k = 0,1,\ldots, LN-1$$



Shifted Discrete Fourier Transforms as discrete representations of the Fourier integral:

L.P. Yaroslavsky, Shifted Discrete Fourier Transforms, In: Digital Signal Processing, Ed. by V. Cappellini, and A. G. Constantinides, Avademic Press, London, 1980, p. 69- 74.

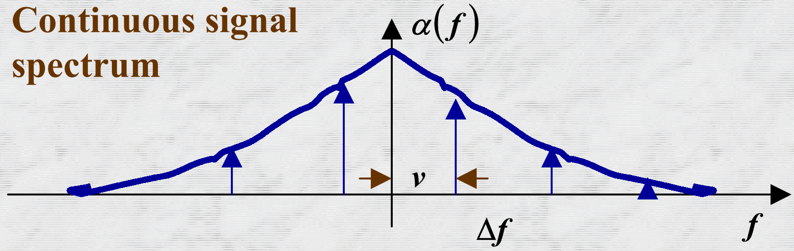
Continuos signal



Sampled signal

$$a(x) = \sum_{k=0}^{N-1} a_k \varphi_{sign\_reconstr}\left(x - (k+u)\Delta x\right)$$

Continuous signal spectrum



Sampled signal spectrum

$$\alpha(f) = \sum_{k=0}^{N-1} \alpha_r \varphi_{spn\_reconstr}\left(f - (r+v)\Delta f\right)$$

Fourier integral ⟹ **Signal and spectrum sampling** ⟹ **Shifted DFT (canonic form)**

$$\alpha(f) = \int_{-\infty}^{\infty} a(x) exp(i2\pi fx) dx \Longrightarrow \alpha_r^{u,v} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \, exp\left(i2\pi \frac{(k+u)(r+v)}{N}\right)$$

$$N = 1/\Delta x \Delta f$$

Direct and inverse Shifted DFTs (reduced form)

$$\alpha_r^{u,v} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \, exp\left(i2\pi \frac{kv}{N}\right) exp\left(i2\pi \frac{(k+u)r}{N}\right)$$

$$\alpha_r^{u,v} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} a_k \, exp\left(-i2\pi \frac{ru}{N}\right) exp\left(i2\pi \frac{k(r+v)}{N}\right)$$

Signal resampling using SDFTs

L. Yaroslavsky, M. Eden, Fundamentals of Digital Optics, Birkhauser, Boston, 1996

Signal $\Rightarrow$ **SDFT(*u*,*v*)** $\Rightarrow$ **ISDFT(*p*,*q*)** $\Rightarrow$ **Sinc-intrerpolated signal shifted by (u-p)**

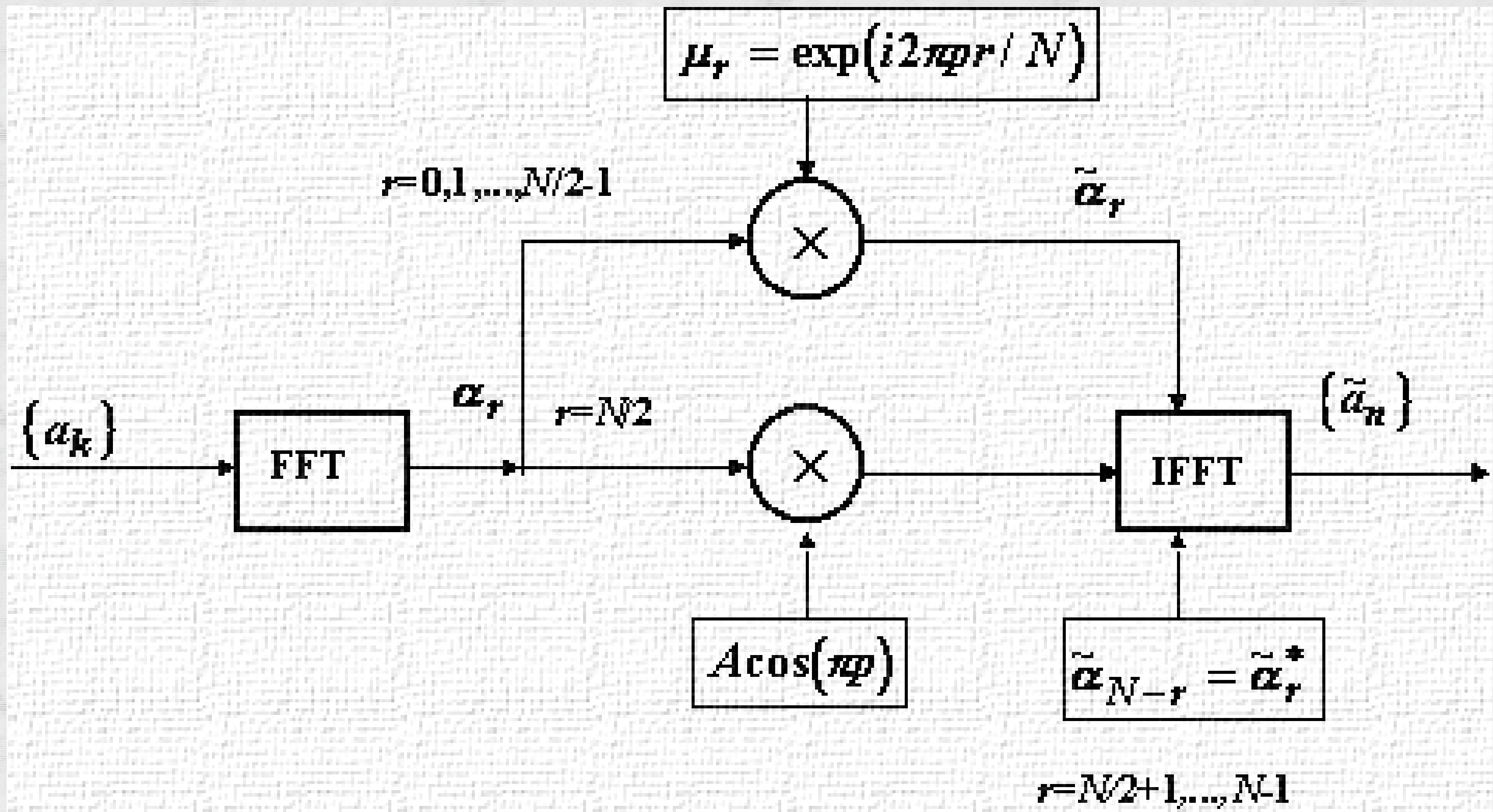
$$\tilde{a}_n^{u/p,v/q} = \frac{1}{\sqrt{N}} \sum_{r=0}^{K-1} \left\{ \alpha_r^{u,v} \exp\left(-i2\pi \frac{rp}{N} \right) \right\} \exp\left(-i2\pi \frac{n(r+q)}{N} \right) =$$

$$\left\{ \sum_{k=0}^{N-1} a_k \exp\left(i\pi k \left(\frac{K-1}{N} + 2v \right) \right) \operatorname{sincd}\left(K; N; (k - n + u - p)\right) \right\} \times$$

$$\exp\left(-i\pi \left(\frac{K-1}{N} + 2q \right) n \right) \exp\left(i\pi \frac{K-1}{N} (u-p) \right)$$

An algorithm of SDFT based signal sinc-interpolation

L. P. Yaroslavsky, Signal sinc-interpolation: a fast computer algorithm, Bioimaging, 4, p. 225-231, 1996



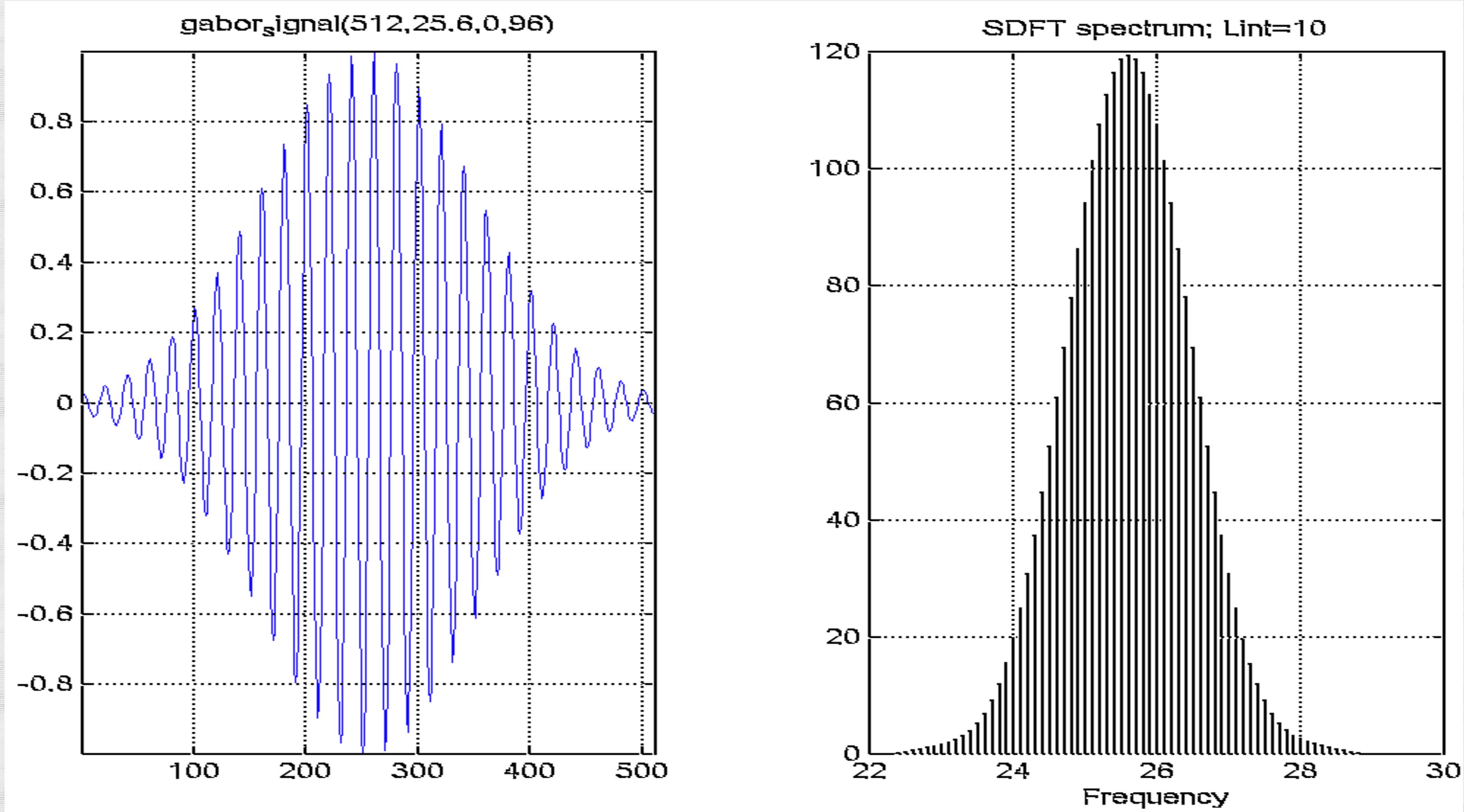
Discrete sinc interpolation versus other convolution based interpolation methods

	SDFT based discrete sinc interpolation	Convolution based interpolation methods Ph. Thevenaz, T. Blu, M. Unser, Interpolation Revisited, IEE Trans. on Medical Imaging, v. 19, No. 17, July 2000
Computational complexity (per output sample)	O(log(Size of the signal frame))	O(Interpolation function support)
Design	Simple: direct spectrum shaping in DFT domain	Requires analytical and numerical optimization
2-D interpolation	Separable & nonseparable	Separable
Interpolation accuracy	Zero interpolation error	Non-zero interpolation error: 1/ O(Interpolation function support)
Invertibility of resampling	Completely invertible	Non-invertible

SDFT versus zero padding sinc-interpolation

	Zero padding method	SDFT based method
Computational complexity (general operations, per output sample) of L-fold zooming signal of N samples with the use of FFT	$O(\log NL)$	$O(\log N)$
Computational complexity (general operations, per output sample) of L-fold zooming signal of N samples in the vicinity of an individual sample	$O(NL\log NL)$ unless FFT pruned algorithms are used	$O(N)$
Computational complexity (general operations, per output sample) for signal shift by a fraction of the discretization interval	$O(L\log NL)$ unless FFT pruned algorithms are used. Shift only by (power of 2)-th fraction of the discretization interval are possible when the most wide spread FFT algorithms are used.	$O(\log N)$ Arbitrary shifts are possible
Zoom factor	Power of 2 for the most widely used FFT algorithms	Arbitrary signal shift; Rational zoom-factor
Memory usage	Requires an intermediate buffer for NL samples	No intermediate buffer is required

Applications:
Spectrum analysis with sub-pixel resolution



Applications: Image rotation: a three-pass algorithm

M. Unser, P. Thevenaz, L. Yaroslavsky, Convolution-based Interpolation for Fast, High-Quality Rotation of Images, IEEE Trans. on Image Processing, Oct. 1995, v. 4, No. 10, p. 1371-1382

Image rotation by an angle θ as a geometrical transformation of signal co-ordinates can be described as a multiplication of signal coordinates (x, y) vector by a rotation matrix:

$$ROT(\theta) = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$$

In order to simplify computations, one can factorize rotation matrix $ROT(\theta)$ into a product of three matrices each of which modifies only one co-ordinate:

(X-shearing Y-shearing X-shearing)

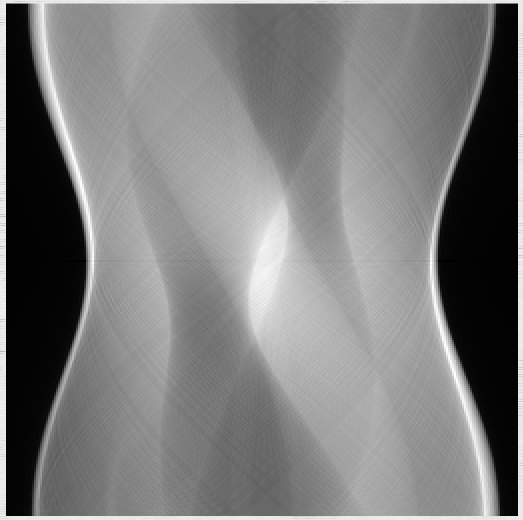
$$ROT(\theta) = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} = \begin{bmatrix} 1 & -tan(\theta/2) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ sin\theta & 1 \end{bmatrix} \begin{bmatrix} 1 & -tan(\theta/2) \\ 0 & 1 \end{bmatrix}$$

Fast signal sinc-interpolation algorithm is ideally suited for signal translation needed for image shearing

Applications: Three pass image rotation with sinc-interpolation

Applications:
Radon Transform and tomosynthesis:
Filtered back projection reconstruction

Radon transform: rotation and directional summation



Tomographic reconstruction: ramp-filtering projections, back projecting, rotation and summation

Applications:
Radon Transform and tomosynthesis:
Fourier reconstruction method

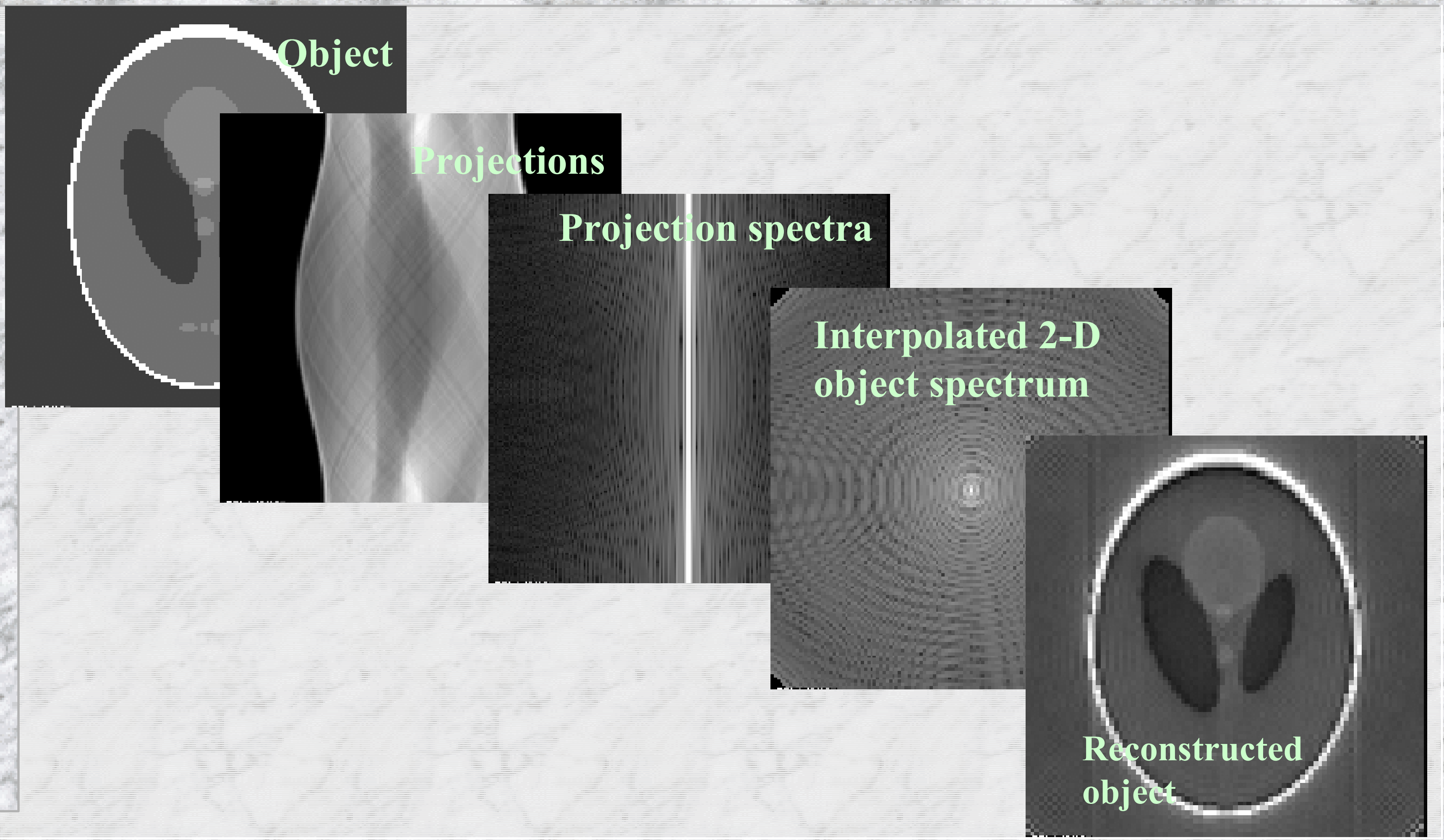


Image geometrical transformations by means of sinc-interpolated image zooming (oversampling)

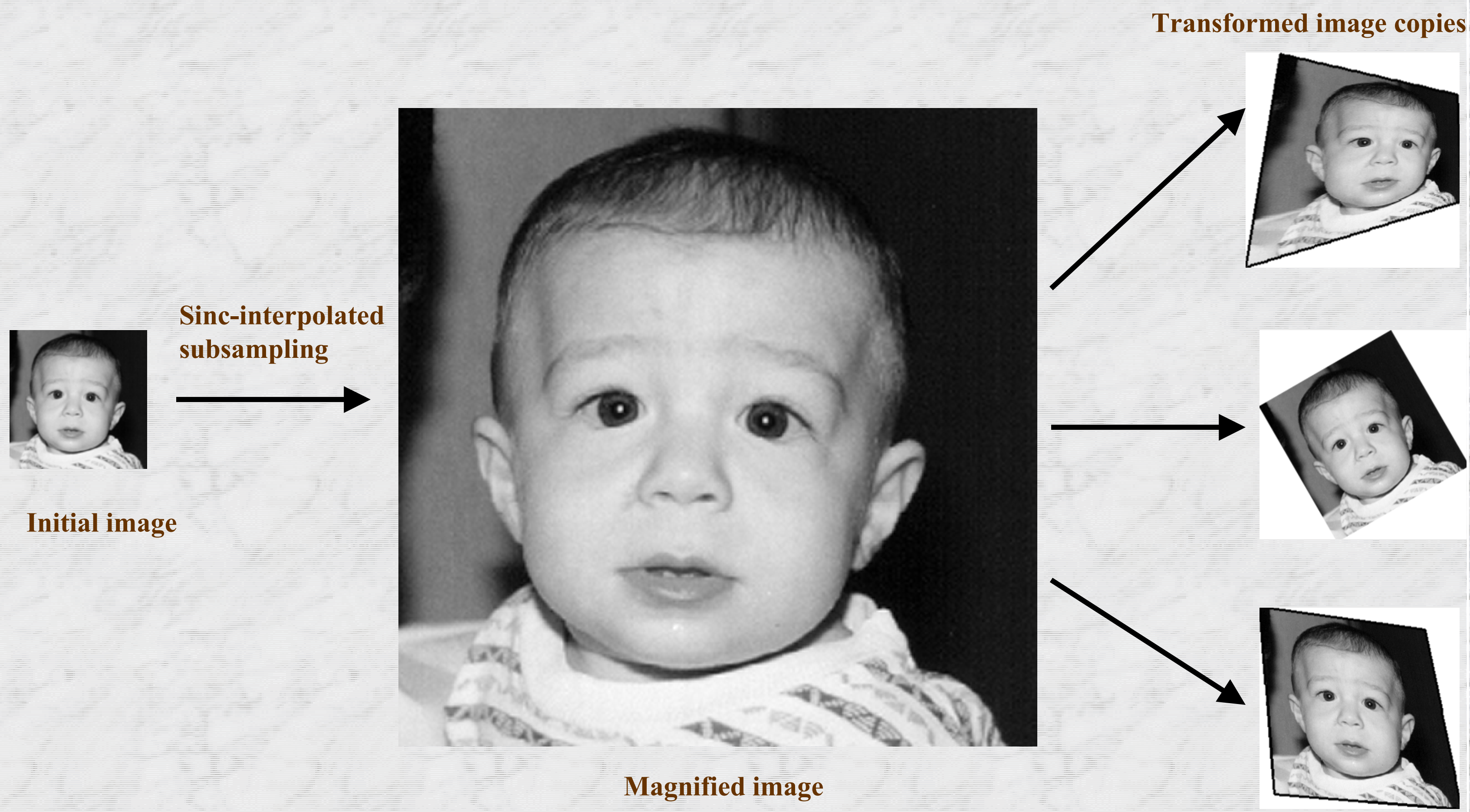
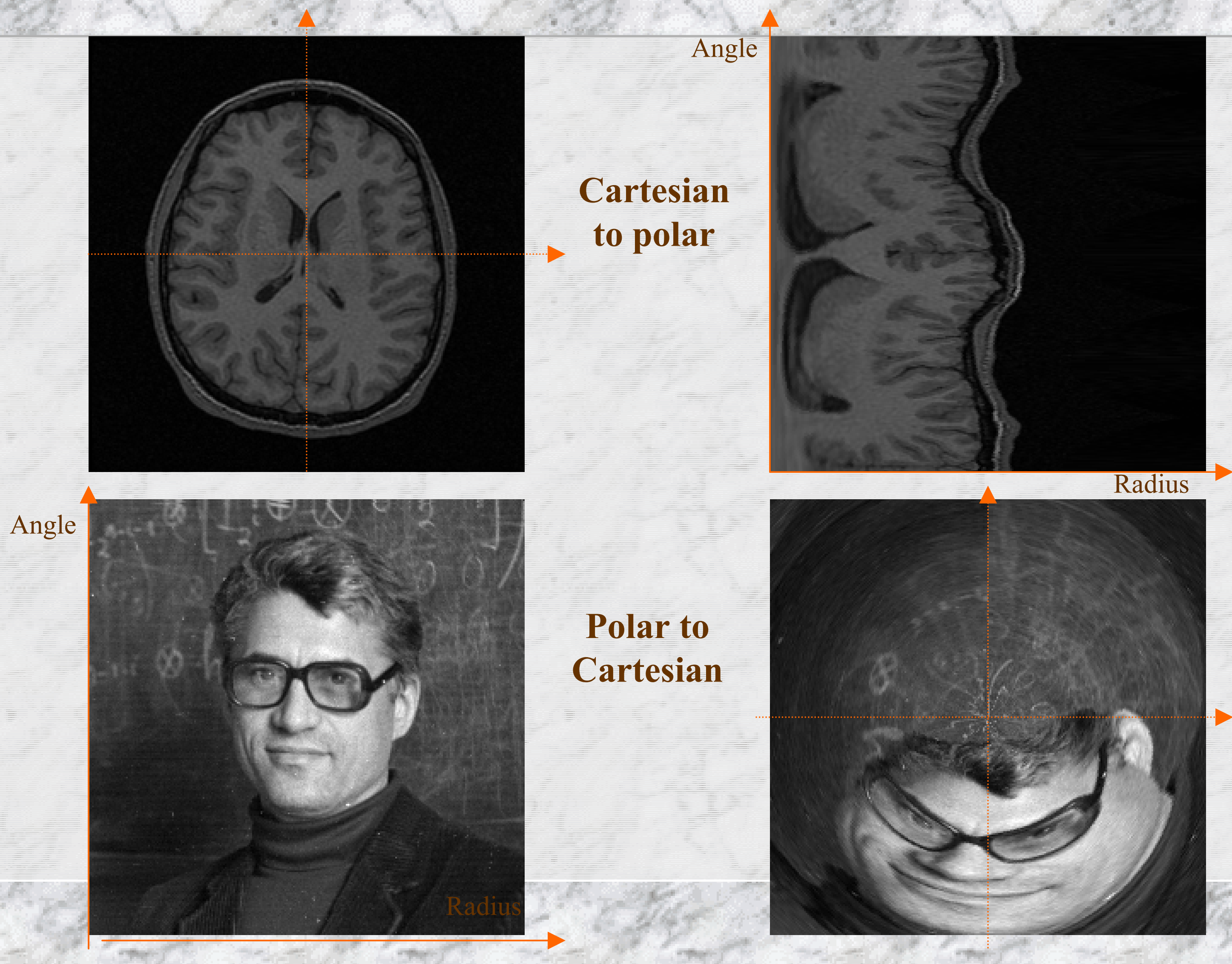
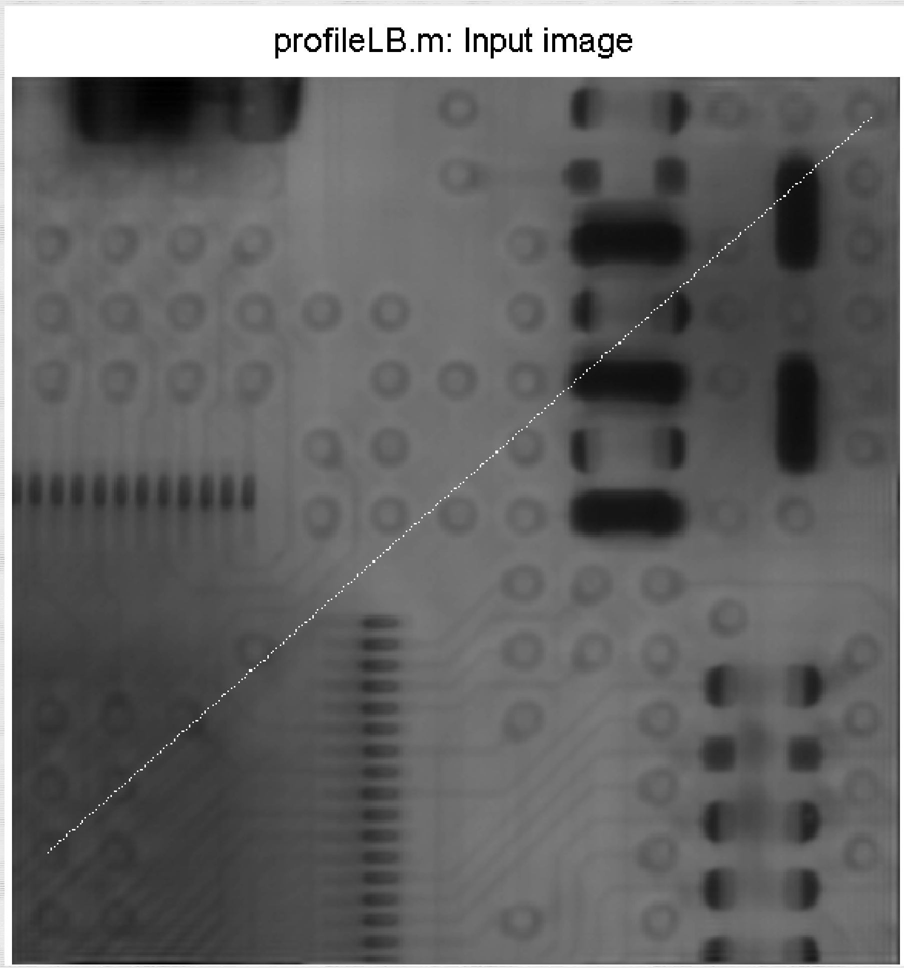
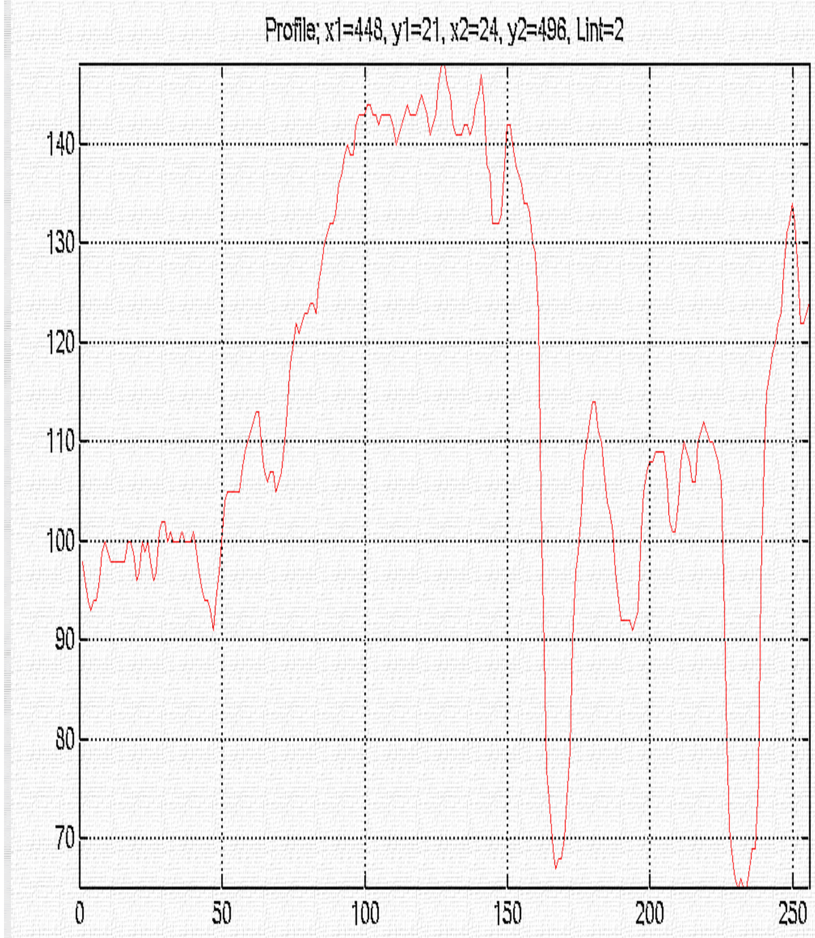


Image geometrical transformations with sinc-interpolation



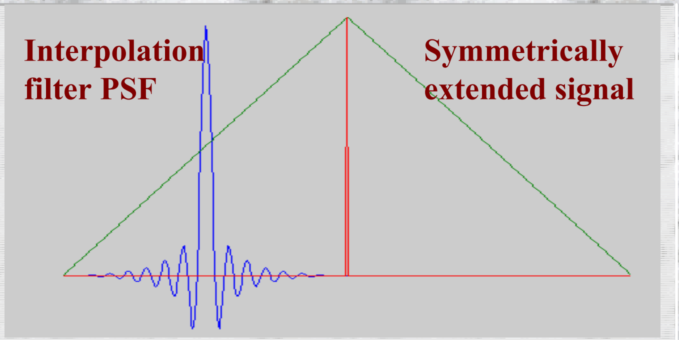
Measuring image profile with sub-pixel resolution





Sincd-interpolation in DCT domain

Purpose: minimization of boundary effects



$$b_k = \frac{1}{\sqrt{2N}} \sum_{r=0}^{2N-1} \beta_r \exp\left(-i2\pi \frac{(k+1/2)r}{2N}\right) =$$

$$\frac{1}{\sqrt{2N}} \left\{ \alpha_0^{DCT} \eta_0 + \sum_{r=1}^{N-1} \alpha_r^{DCT} \left[\eta_r \exp\left(-i\pi \frac{(k+1/2)}{N} r\right) + \eta_r^* \exp\left(i\pi \frac{(k+1/2)}{N} r\right) \right] \right\}$$

For *u*-shift

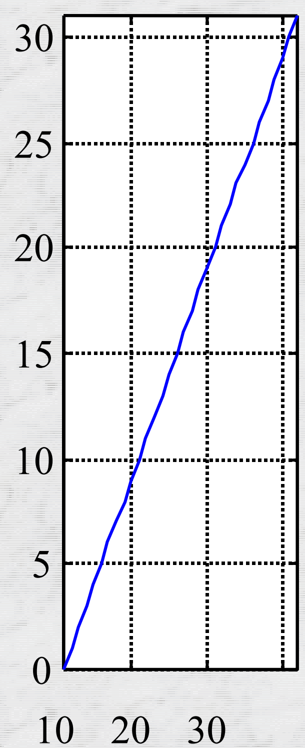
$$h_k = \begin{cases} \frac{1}{\sqrt{N}} \sum_{s=0}^{N-1} \varphi_s \exp\left(-i2\pi \frac{ks}{N}\right); & k = 0,1,...N-1 \\ 0; & k = N, N+1,...2N-1 \end{cases} \qquad \varphi_s = \begin{cases} \exp(i2\pi us/N); & s = 0,1,...,N/2-1 \\ \cos(2\pi us/N); & s = N/2 \\ \varphi_{N-s}^*; & s = N/2+1,...,N-1 \end{cases}$$

$$\eta_{2r} = \exp\left(i2\pi \frac{ur}{N}\right)$$

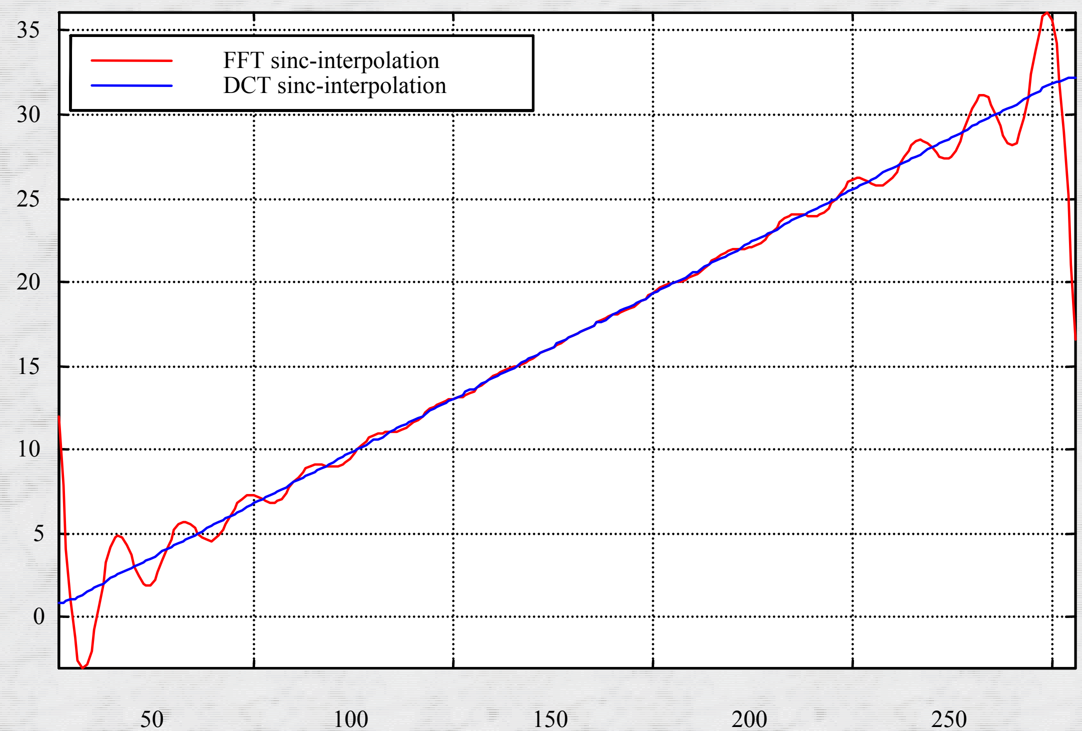
$$\eta_{2r+1} = \frac{1}{\sqrt{2}N} \sum_{k=0}^{N-1} \sum_{s=0}^{N-1} \varphi_s \exp\left(-i2\pi \frac{ks}{N}\right) \exp\left(i2\pi \frac{k(2r+1)}{2N}\right) = \frac{1}{\sqrt{2}N} \sum_{k=0}^{N-1} \sum_{s=0}^{N-1} \varphi_s \exp\left(i2\pi \frac{k(2r-2s+1)}{2N}\right)$$

Pruned DFT is useful: first half of the sequence is zero, only odd terms of DFT are to be computed

Sincd-interpolation in DCT domain

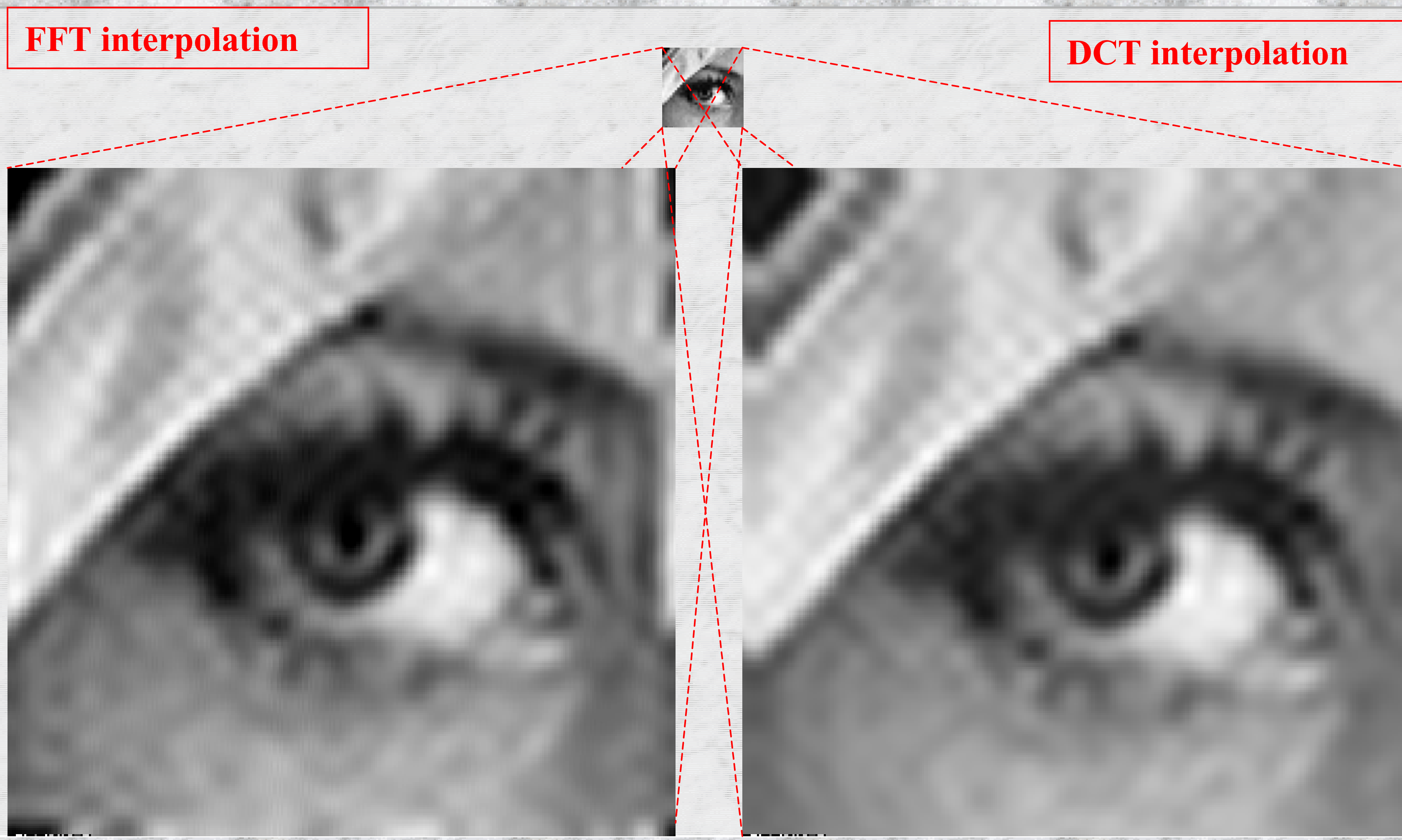


Initial signal



8x sinc-interpolated signals

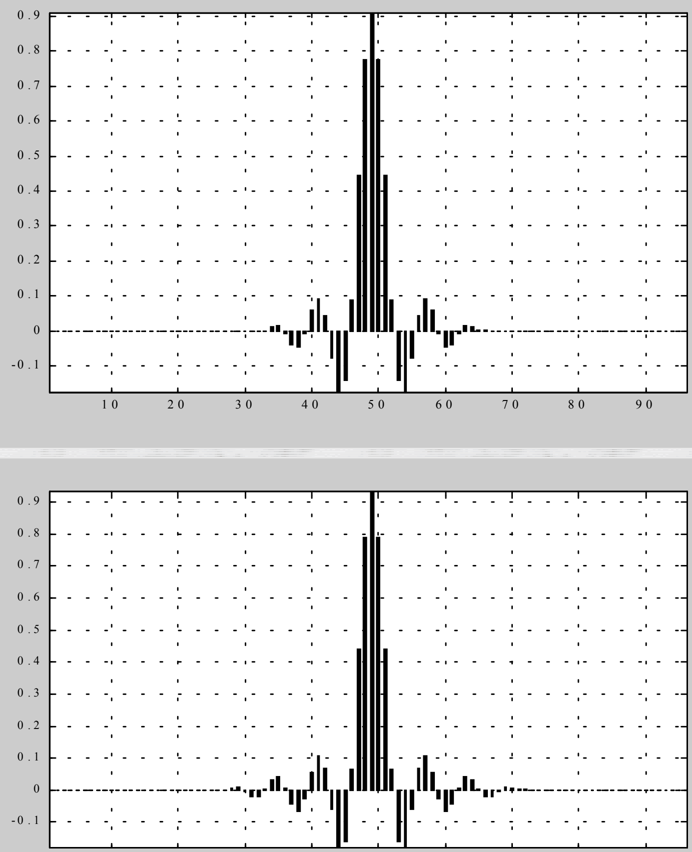
2-D separable sincd-interpolation in DCT domain



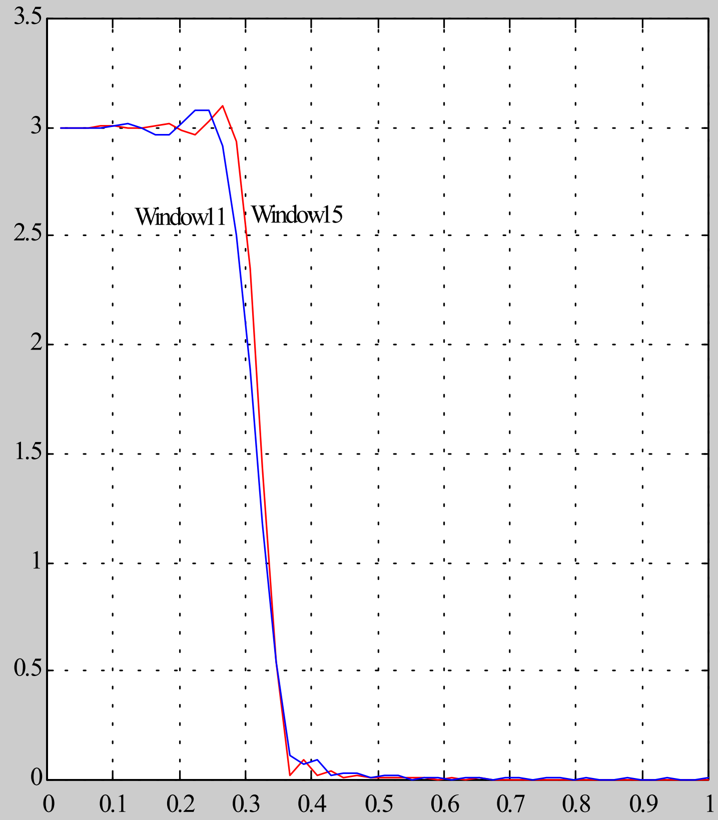
Signal resampling in sliding window: Summary

- ✓Good approximation to sincd-interpolation
- ✓Simple design in DFT domain
- ✓Irregular -to-regular sampling raster resampling is possible
- ✓Can naturally be combined with signal denoising and restoration
- ✓Local adaptive interpolation is possible
- ✓Implementation of inseparable interpolation kernel is easy
- ✓Computational complexity O(WindowSize) per output signal sample (comparable to that of spline interpolation)

Signal sinc-interpolation in sliding window: Impulse and frequency responses

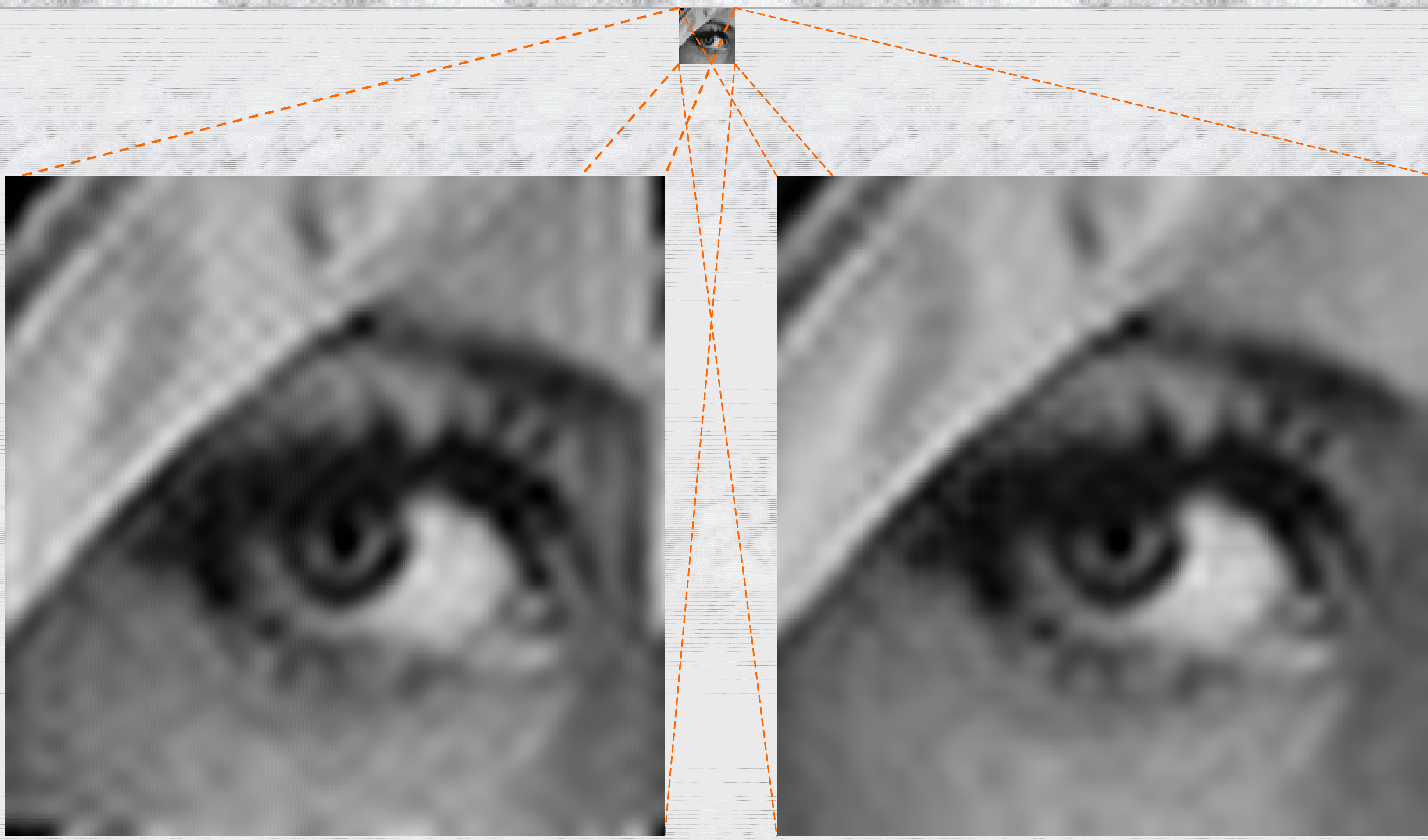


Interpolation functions for window size 11 and 15 samples



Interpolation filter frequency responses: signal 3x zoom

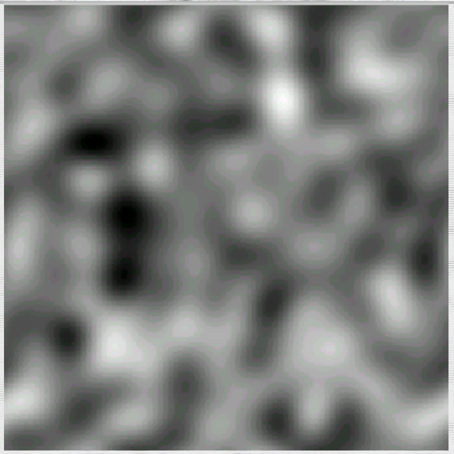
Image zoom: global versus sliding window sinc-interpolation



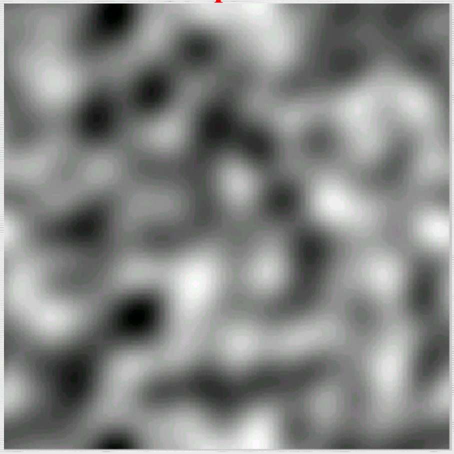
Global sinc-interpolation

Sliding window sinc-interpolation

Arbitrary image mapping in sliding window



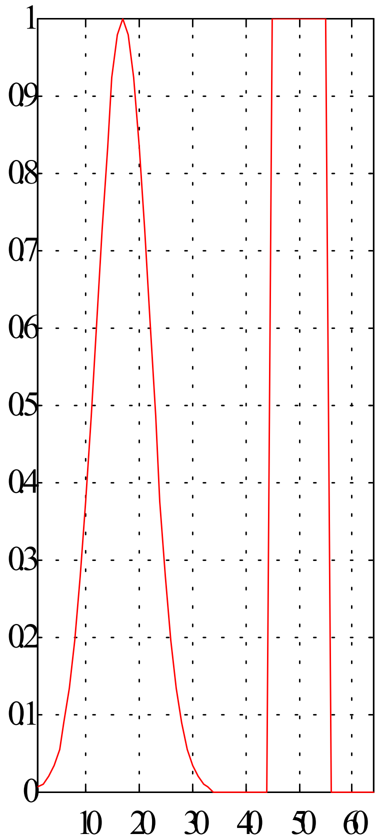
MapX



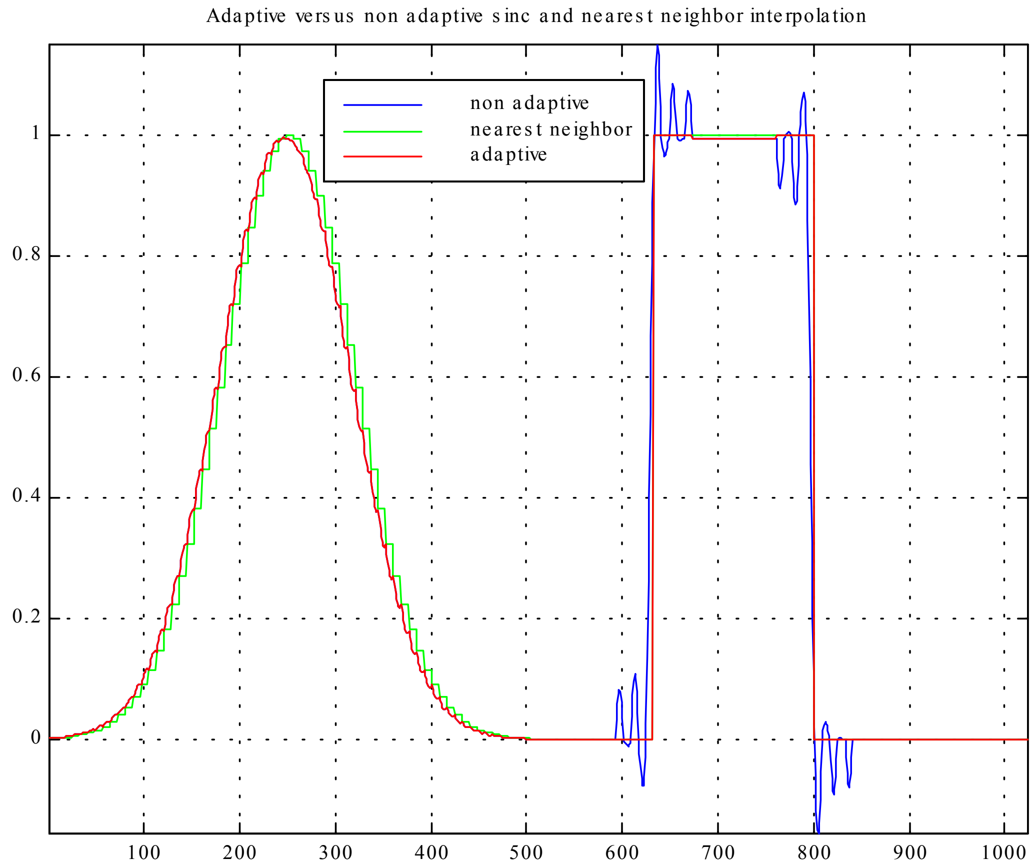
MapY

lcmapping_arbitr(len,(mapX+i*mapY)/1.5,8,8)

Sliding window sinc-interpolation: non-adaptive vesrus adaptive



Initial signal



Interpolated signal (8xZoom)

Sliding window sinc-interpolation: non-adaptive vesrus adaptive

non-adaptive

adaptive

Irregular-to-regular raster image resampling and denoising in sliding window

Additional bibliography on FFT based and spline based interpolation

- D. Fraser, Interpolation by the FFT revisited – An experimental evaluation, IEEE Trans. ASSP-37, p. 665-676, May 1989
- T.J. Cavichi, DFT time domain Interpolation, Proc. Inst.Elec.Eng.-F, vol. 139, pp. 207-211, 1992
- M. Unser, A. Aldroubi, M. Eden, Plynomial spline signal approximations: filter design and asymptotic equivalence with Shannonäs sampling theorem, IEEE Trans. IT, v. 38, pp. 95-103, Jan. 1992
- Ph. Thevenaz, Th. Blu, M. Unser, Interpolation Revisited, IEEE Trans. on Medical Imaging, V. 19, No. 7, July 2000