

L. Yaroslavsky. Course 0510.7211 "Digital Image Processing: Applications"

Lect. 8 Methods of image filtering and resampling in signal and transform domain

Filtering in signal domain. General linear filter: $\mathbf{b} = \mathbf{H}\mathbf{a}$;

Computational complexity: $O(\text{Size}(\mathbf{H}))$ per pixel.

Recursive filtering: $b_k = \sum_{n=0}^{N^{(l)}-1} h_n^{(l)} a_{k-n} + \sum_{s=1}^{N^{(r)}} h_s^{(r)} b_{k-s}$

2-D separable filtering $b_{k,l} = \sum_{m=0}^{M-1} h_m^{(2)} \sum_{n=0}^{N-1} h_n^{(1)} a_{k-n,l-m}$. Computational Complexity

Cascade, parallel and parallel recursive filtering.

Boundary effects.

Filtering in transform domain. Scalar filters: $\mathbf{b} = \mathbf{T}^{-1} \mathbf{H}_d \mathbf{T} \mathbf{a}$; $\beta_r = \eta_r \alpha_r$

Computational complexity: $O(\log(\text{Size}(\mathbf{a})))$ per pixel with the use of Fast Transforms.

Applications of DFT and SDFTs for signal spectral analysis, convolution.

Spectral analysis and signal convolution through DFT, SDFT and DCT. Boundary effects and methods to deal with them: "zero"-padding, signal windowing, symmetrisation.

Signal interpolation (re-sampling) and image geometrical transformations.

Signal interpolation computational convolution based methods:

Nearest neighbour (zero order) interpolation: $\tilde{a}_{\text{int}}^M = \text{conv}[\text{kron}(\tilde{a}, \delta(M)), \text{rect}(M)]$;

$\text{rect}(M) = \text{ones}(M)$; $\delta(M) = [1, 0, 0, \dots, 0]$

Linear (bilinear) interpolation: $\tilde{a}_{\text{int}}^M = \text{conv}[\text{conv}[\text{kron}(\tilde{a}, \delta(M)), \text{rect}(M)], \text{rect}(M)]$

Spline interpolation: $\tilde{a}_{\text{int}}^M = \text{conv}[\text{conv} \dots [\text{conv}[\text{kron}(\tilde{a}, \delta(M)), \text{rect}(M)], \text{rect}(M)]]$.

Cardinal splines interpolators

Discrete sampling theorem and optimality of the discrete sinc-interpolation:

$$\tilde{a}_{\text{int}}^M = \text{conv}[\text{kron}(\tilde{a}, \delta(M)), \text{sincd}(\pm 1; N; 1/M)]$$

Sinc-interpolation by zero padding the DFT spectrum and its computational complexity: $O(NL \log NL)$.

Sinc-interpolation by SDFT:

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

$$\sum_{k=0}^{N-1} a_k \exp\left[i2\pi \frac{k}{N} \left(v + \frac{K-1}{2}\right)\right] \text{sincd}(K; N; k-n+u-p) \exp\left[-i2\pi \frac{n}{N} \left(q + \frac{K-1}{2}\right)\right] \exp\left(i\pi \frac{K-1}{N} (u-p)\right)$$

$$a(x) = \sum_{k=0}^{N-1} a_k \text{sincd}(K; N; x/\Delta x - k); \quad \text{sincd}(K; N; x) = \frac{\sin[\pi Kx/N]}{N \sin[\pi x/N]}$$

$$\text{sincd}(\pm 1; N; x) = [\text{sincd}(N-1; N; x) + \text{sincd}(N+1; N; x)]/2$$

Computational complexity $O(NL \log N)$. Flexibility and memory requirements.

Image geometrical transformations by means of image zooming-up with sinc-interpolation.

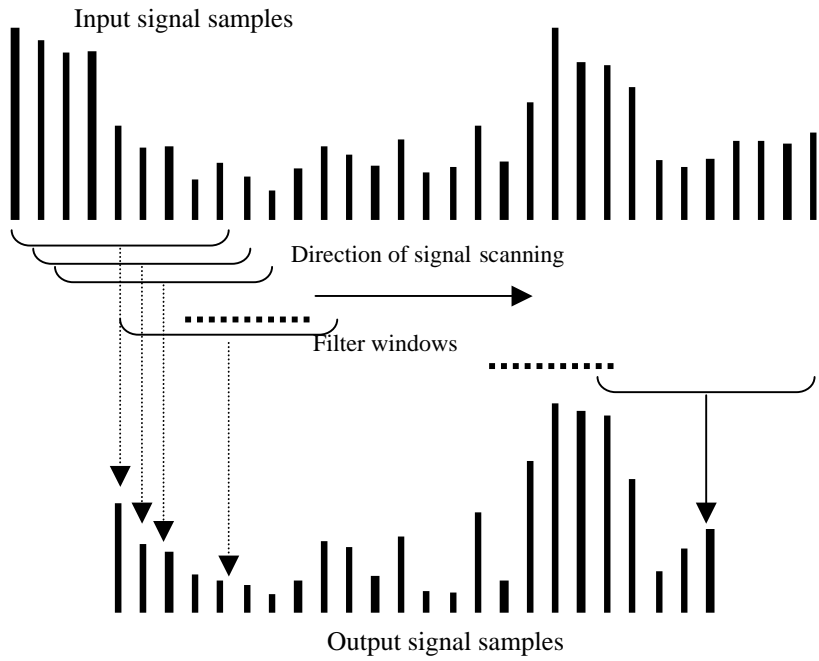
Image rotation with sinc-interpolation: a three-pass rotation algorithm

(X-shearing Y-shearing X-shearing).

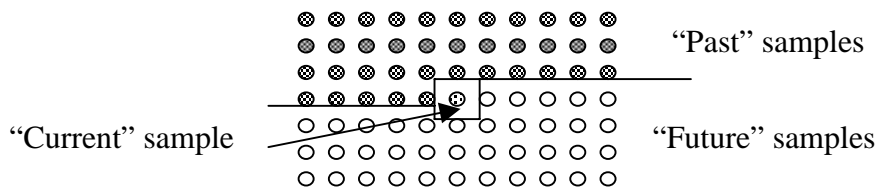
$$\text{ROT}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} = \begin{bmatrix} 1 & -\tan(\theta/2) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \sin \theta & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tan(\theta/2) \\ 0 & 1 \end{bmatrix}$$

Problems for self-testing

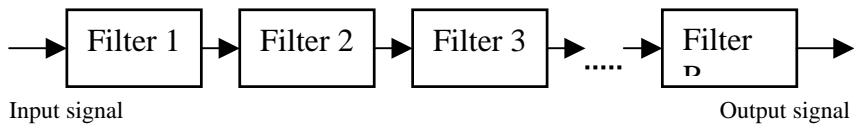
1. Compare advantages and disadvantages of filtering in signal and transform domains.
2. What methods exist to accelerate image filtering in signal domain?
3. Describe the use of DFT for signal spectral analysis; describe the role of boundary effects and spectral windows.
4. Compare interpolation methods in terms of accuracy and computational complexity.
5. Derive SDFT based signal sinc-interpolation . Compare zero padding and SDFT methods.
6. Explain using SDFT sinc-interpolation for 3-pass image rotation algorithm.



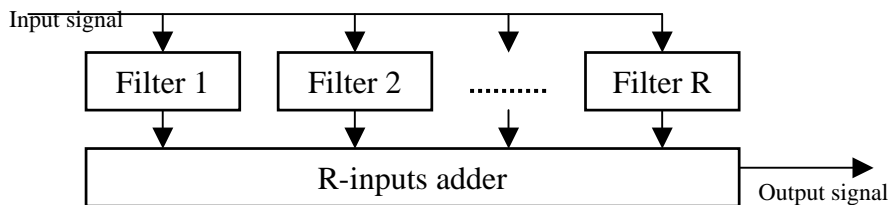
Signal domain filtering in sliding window



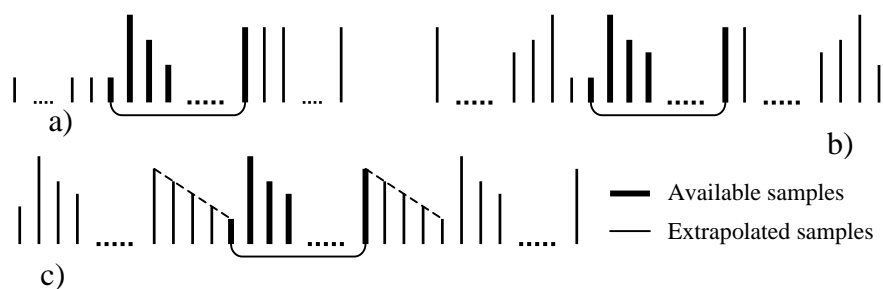
"Past" and "future" samples on a rectangular sampling raster at 2-D row-wise filtering



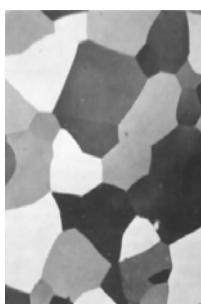
The principle of cascade filtering



The principle of parallel implementation of filtering



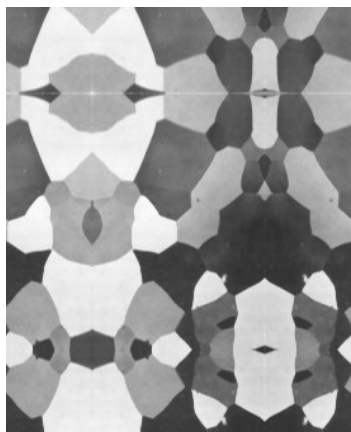
Three signal extrapolation methods that secure the absence of signal discontinuity on the borders: a) - repeating border samples; b) - signal mirroring; c) - interpolation between border samples in the assumption of signal periodical replication



Original image



Extrapolation by replicating border samples



Extrapolation by mirroring

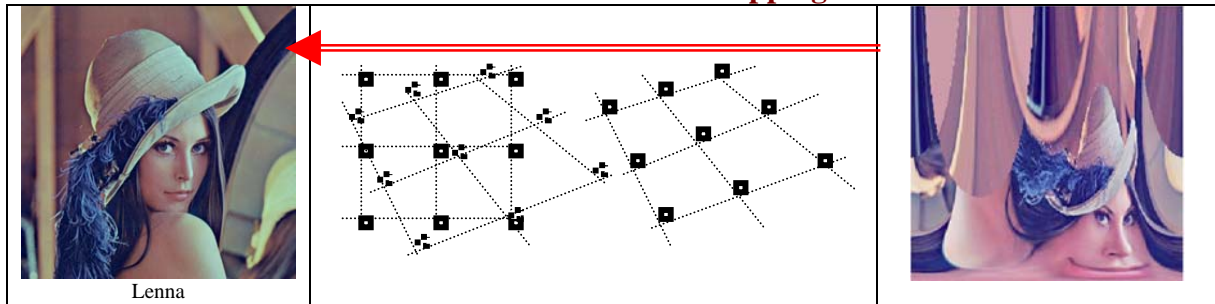


Linear interpolation between borders

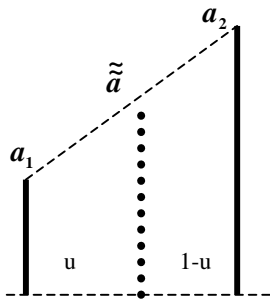
Three methods for extrapolating images without discontinuities on its borders

Digital signal/image geometrical transformations and signal resampling

The principle of image geometrical transformations: backward co-ordinate mapping

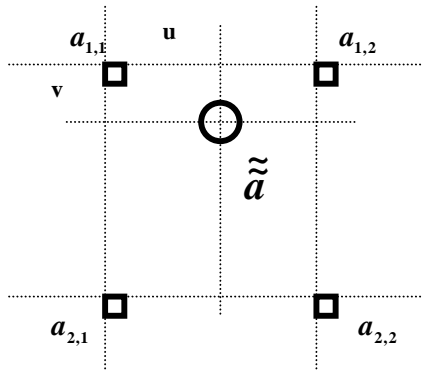


Examples of image geometrical transformations



$$\tilde{a} = ua_2 + (1-u)a_1$$

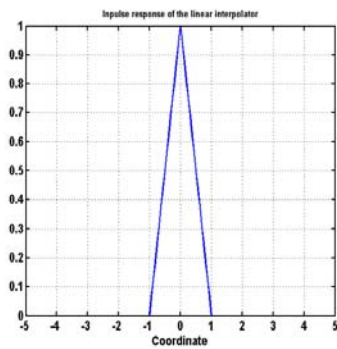
a)



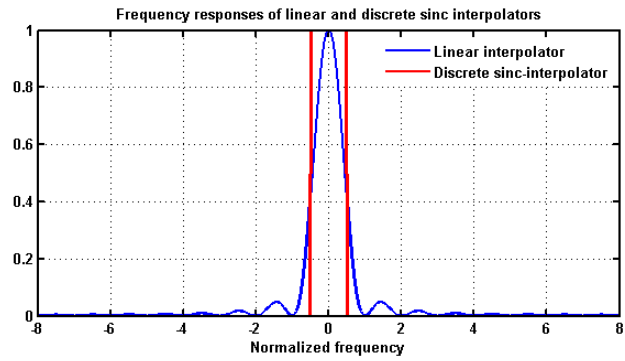
$$\tilde{a} = uva_{2,2} + (1-u)va_{2,1} + u(1-v)a_{1,2} + (1-u)(1-v)a_{1,1}$$

b)

The principle of linear (a) and bilinear interpolation (b)

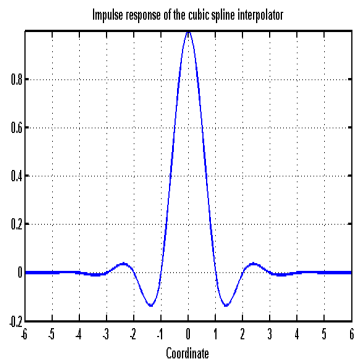


a)

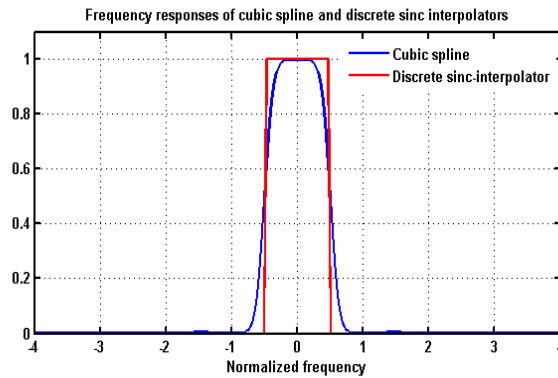


b)

Impulse response (a) and frequency response of the linear interpolation (b)



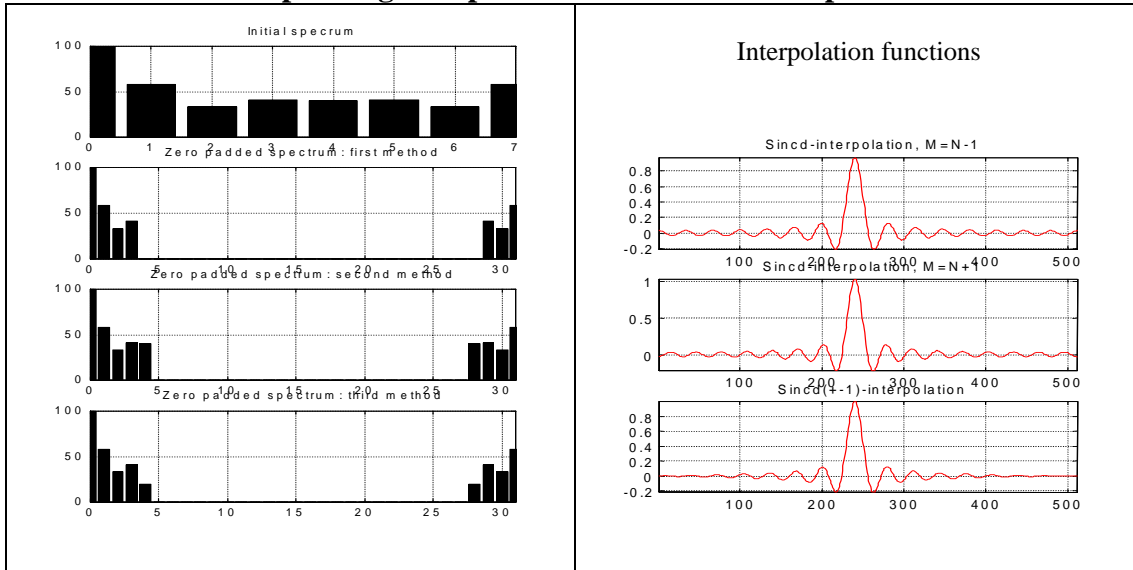
a)



b)

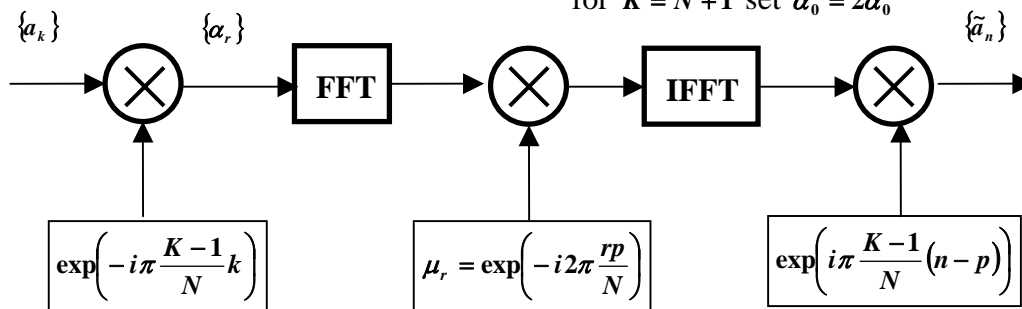
Cardinal cubic spline interpolation kernel (a) and frequency response (b)

Zero-padding interpolation method and interpolation kernels

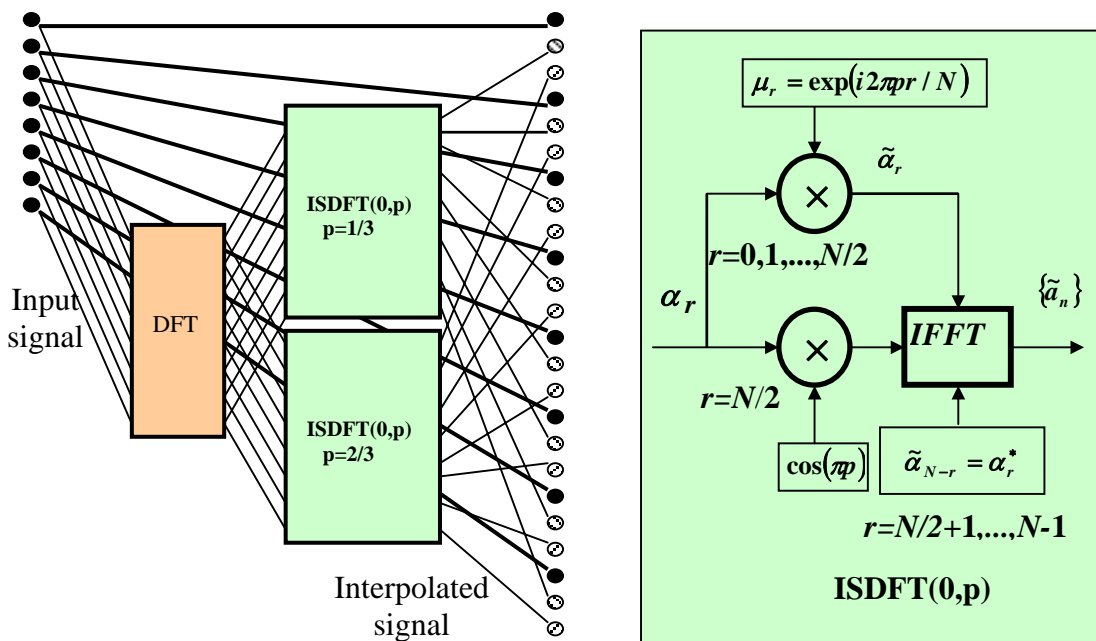


In IFFT, for $K = N - 1$ set $\alpha_{N-1} = 0$;

for $K = N + 1$ set $\alpha_0 = 2\alpha_0$



The principle of fast discrete sinc-interpolation



SDFT p -shift algorithm for fast discrete sinc-interpolation (3x - subsampling)

8x image zoom: comparison of image interpolation methods

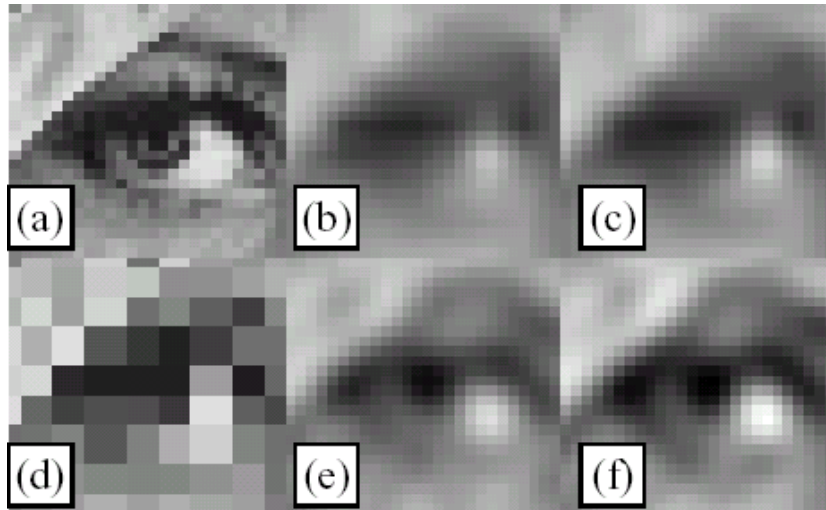
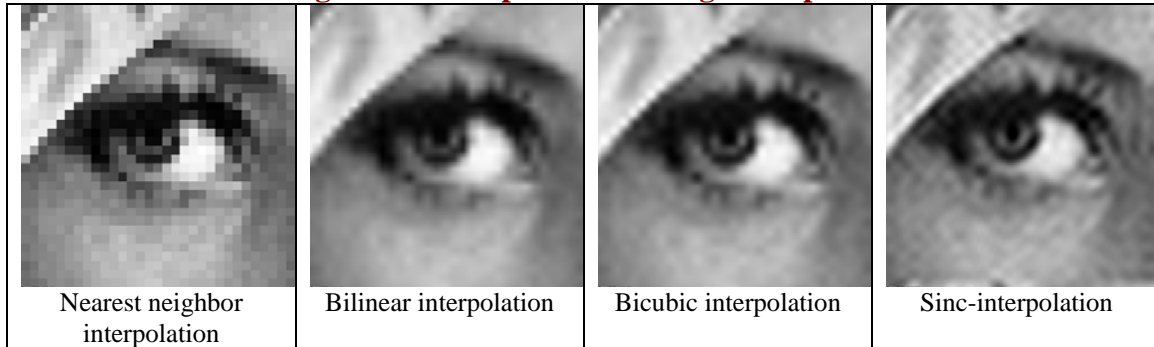


Image (a) is reduced with reduction factor $77/256$ and then magnified with magnification factor $256/77$. b) – bilinear interpolation; c) – bicubic interpolation; d) – nearest neighbor interpolation; e) – sincd-interpolation 11×11 ; f) – sincd interpolation 31×31

Comparison of zero-padding and SDFT based sinc-interpolation algorithms.

	Zero padding method	SDFT based method
Computational complexity (general operations) of L -fold zooming signal of N samples with the use of FFT	$O(NL \log NL)$	$O(NL \log N)$
Computational complexity (general operations) of L -fold zooming signal of N samples in the vicinity of an individual sample (as, for instance, in locating position of signal maximum with sub-pixel accuracy)	$O(NL \log NL)$, unless FFT pruned algorithms are used	$O(NL)$
Computational complexity (general operations) for signal shift by a fraction of the discretization interval	$O(NL \log NL)$, unless FFT pruned algorithms are used; shifts only by (power of 2)-th fraction of the discretization interval are possible when the most wide spread FFT algorithms are used.	$O(N \log N)$; arbitrary shifts are possible
Zoom factor	Power of 2 for the most widely used FFT algorithms	Arbitrary
Memory usage	Requires an intermediate buffer for NL samples	Does not need an intermediate buffer

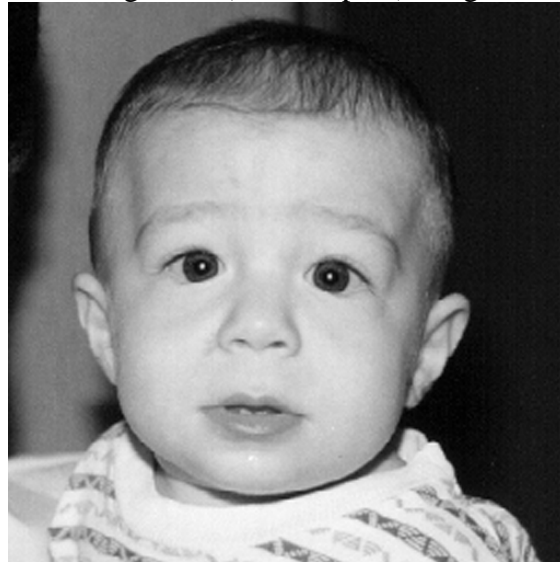
Image geometrical transformations by means of oversampling (zooming)

Initial image

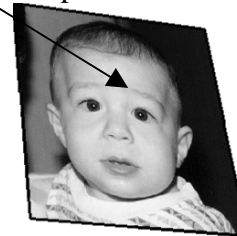
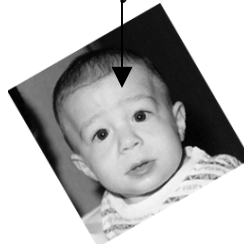


Image zoom with
sinc-interpolation

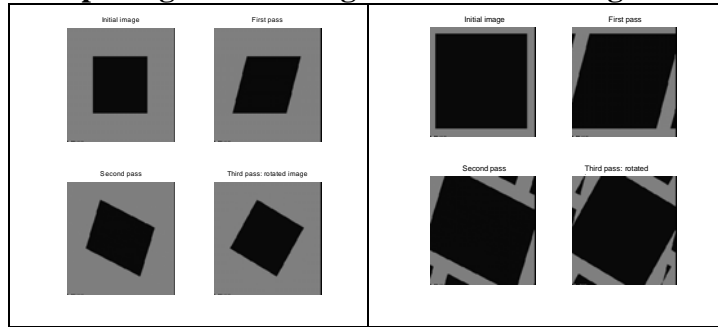
Magnified (oversampled) image



Transformed images obtained by nearest neighbour interpolation



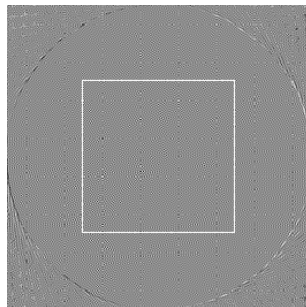
3-step image rotation algorithm and aliasing effects



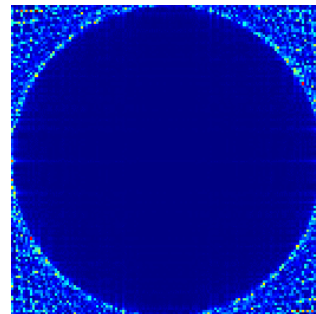
Initial image



18*20=360° rotated image

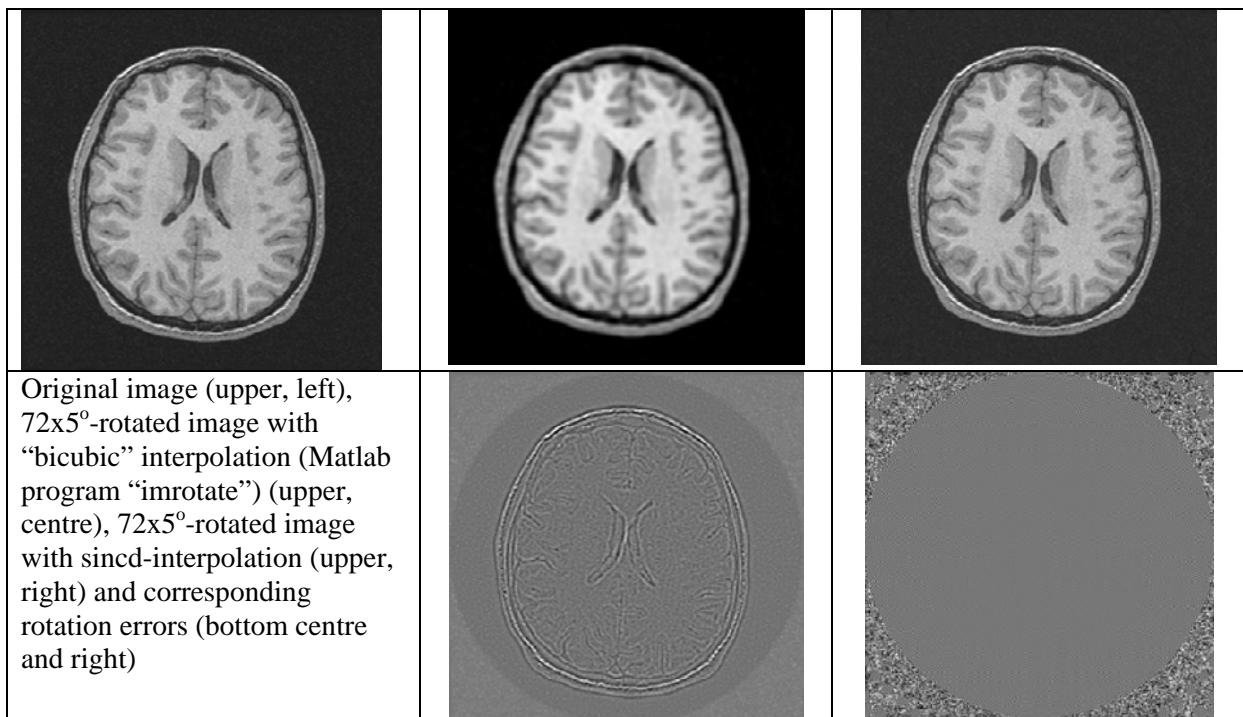


Rotation error



Spectrum of the error

Image rotation with SDFIT p-shift sinc-interpolation algorithm



Comparison of “bicubic” and discrete sinc-interpolation