Lect. 8 Applications of Discrete Fourier Transform

1.Signal spectral Fourier analysis.

Spectrum estimation and spectral windows.
DCT and DST as a spectrum analysers.

2.Signal convolution

Filtering in spectral domain: fast convolution:

$$\left.\begin{matrix}\{a_k\}\xrightarrow{DFT}\{\alpha_r\}\\ \{b_k\}\xrightarrow{DFT}\{\beta_r\}\end{matrix}\right\}\Rightarrow\{\alpha_r\beta_r\}\xrightarrow{IDFT}\frac{1}{\sqrt{N}}\left\{\sum a_n b_{(k-n)\bmod N}\right\}$$

Per sample complexity for signal of N samples: $3(\log_2 N-1)+4/N$ additions, $2(\log_2 N-3)+8/N$ multiplications versus $WSz-1$ additions and WSz multiplications for convolution in signal domain with a filter PSF of WSz samples
Boundary effects and methods of signal extension to compensate them.
Convolution with DCT.

3. Signal interpolation.

Sinc-interpolation by spectrum zero padding:

$$a(x)=\sum_{k=0}^{N-1}a_k\,\mathbf{sincd}(M;N;x/\Delta x-k);\qquad \mathbf{sincd}(M;N;x)=\frac{\sin[\pi Mx/N]}{N\sin[\pi x/N]};$$

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

Drawbacks: inflexibility and computational complexity $O(LN\log LN)$ for L-fold interpolation of a signal of N samples
Sinc-interpolation by SDFT: **OUTSignal=ISDFT$_{p,q}$(SDFT$_{u,v}$(INPUTSignal))**

$$a((k+p-u)\Delta x)=a_n^{0/p;v/q}=\left\{\sum_{k=0}^{N-1}a_k\,\mathbf{sincd}(M;N;k-n+u-p)\right\}\exp\left(-i\pi\frac{M-1}{N}p\right)$$

2-D signal interpolation: separable versus inseparable algorithms

Interpolation and image geometrical transformations.

Convolution based methods of signal resampling/interpolation:
Nearest neighbour interpolation: $a_k^{\text{int}}=\{a_n\}*\{rect(n-k/L)\}$
Linear (for 2-D – bilinear) interpolation: $a_k^{\text{int}}=\{a_n\}*\{rect(n-k/L)\}*\{rect(n-k/L)\}$
Cubic (for 2-D bicubic) spline interpolation:

$$a_k^{\text{int}}=\{a_n\}*\{rect(n-k/L)\}*\{rect(n-k/L)\}*\{rect(n-k/L)\}$$

Sinc interpolation: $a_k^{\text{int}}=\{a_n\}*\{\mathbf{sincd}(\pm 1,N,(n-k/L))\}$

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

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

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

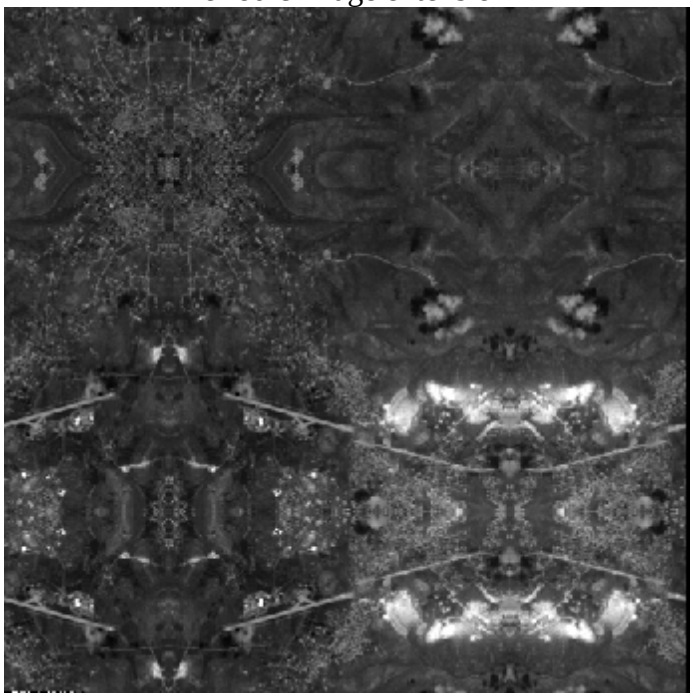
3-D rotation.

Problems:

1. Explain the use of DFT and DCT for signal spectral analysis; describe the role of boundary effects and spectral windows.
2. Explain the use of DFT for signal convolution, boundary effects and how to get over them?
3. Describe methods of discrete signal interpolation, their accuracy and computational complexity. Why sinc interpolation is a gold standard for linear interpolation of sampled data?
4. Describe fast sinc-interpolation algorithm and compare it with zero-padding method of sinc-interpolation
5. Describe 3-step image rotation algorithm with sinc-interpolation.

Homework: using matlab program **imrotate.m**, compare nearest neighbour, bilinear and bicubic interpolation for successive image rotations from 0 to 360^0 carried out in several steps

Periodic image extension



Mirror image extension

IMAGE GEOMETRICAL TRANSFORMATIONS and SIGNAL INTERPOLATION

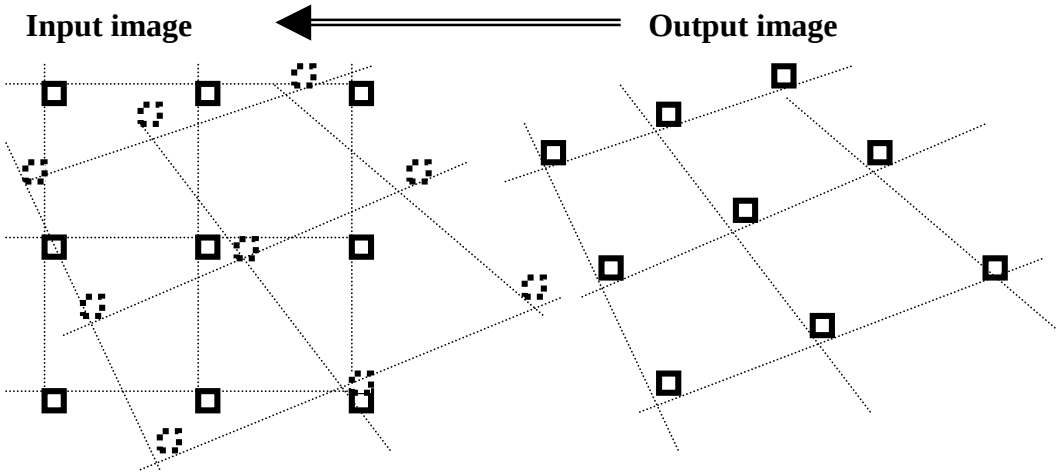
Lenna

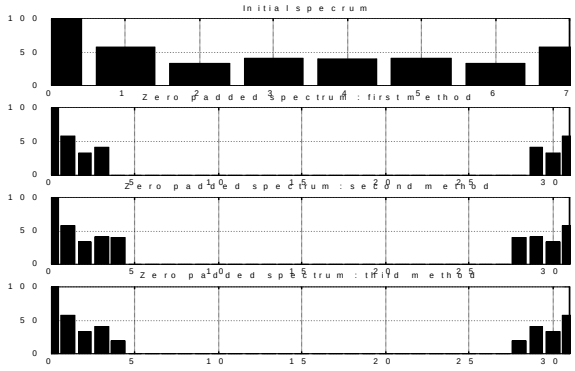
y=x+3sin(2pix/32); mapping(lenna,y,y)

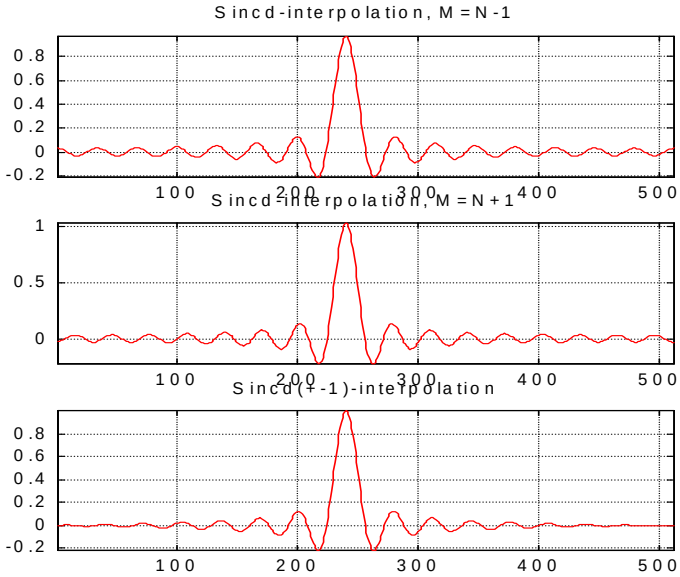
y=x+3cos(2pix/32); mapping(lenna,y,y)

y=x+20exp(((x-128)^2/1024); mapping(lenna,y,y)

Principle of image geometrical transformation: backward co-ordinate mapping



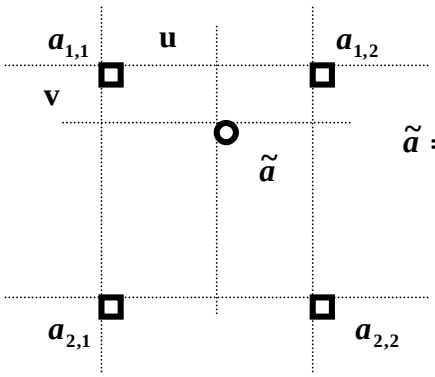




Three methods of sinc-interpolation for even number of signal samples

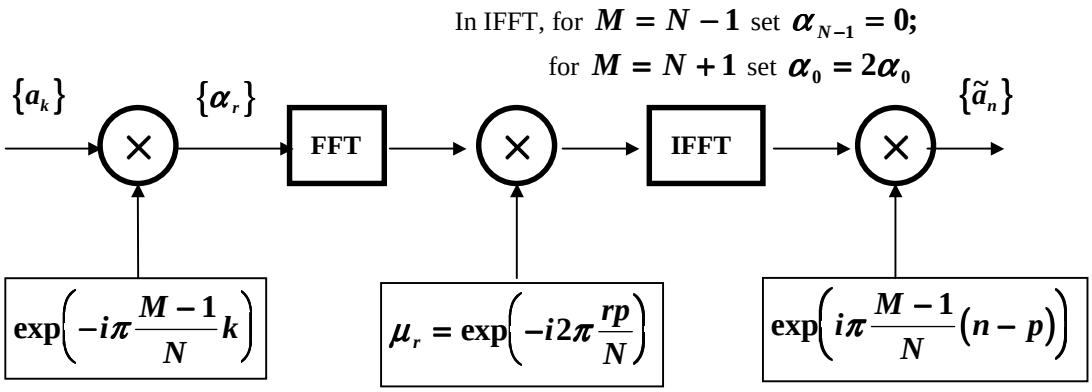
Nearest neighbour interpolation	Bilinear interpolation
Bicubic interpolation	Sinc interpolation

Nearest neighbour, bilinear, bicubic and sinc interpolation for image zooming (zoom x8)

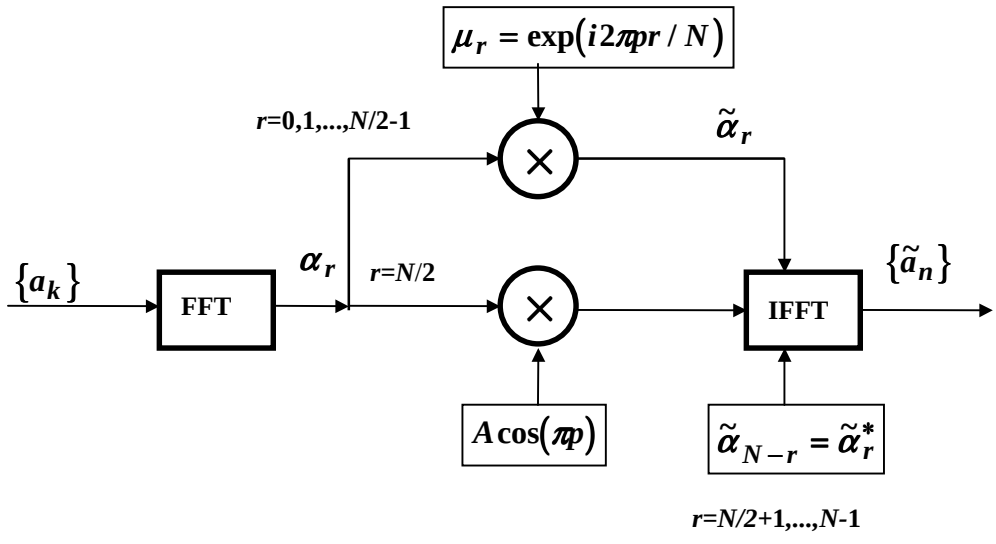


$$\tilde{a} = uva_{1,1} + (1-u)va_{1,2} + u(1-v)a_{2,1} + (1-u)(1-v)a_{2,2}$$

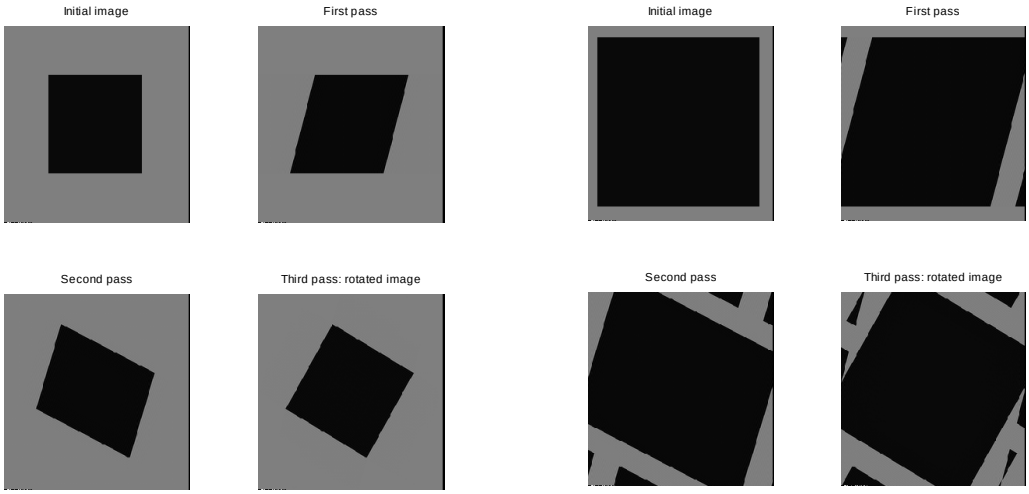
Bilinear interpolation



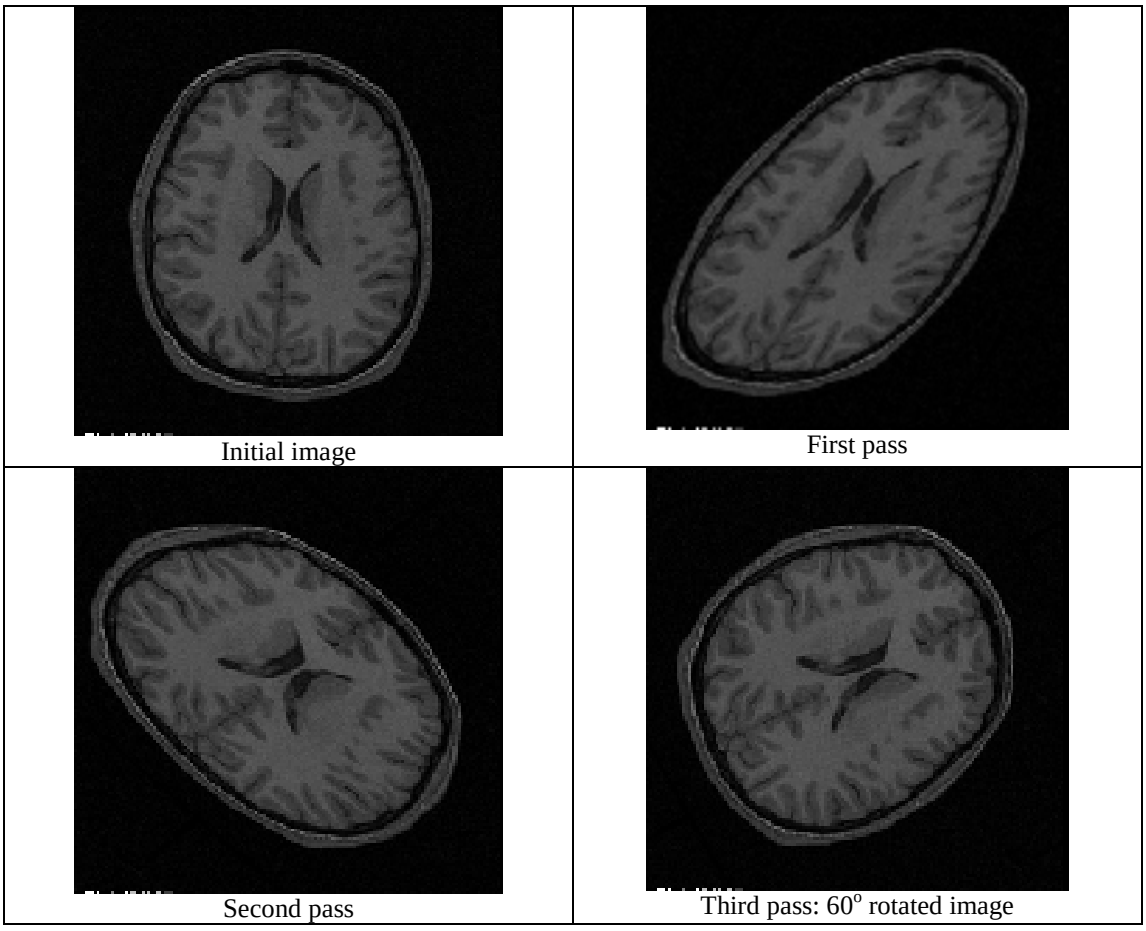
An algorithm for discrete sinc-interpolation



A modified algorithm for discrete sinc-interpolation



3-pass image rotation: aliasing effects



3-pass image rotation: an example