

Digital Image

(B4M33DZO)

Lecture 2:

Fourier Transform

<https://cw.fel.cvut.cz/wiki/courses/dzo/start>

Ondřej Drbohlav & Daniel Sýkora

Department of Cybernetics

Faculty of Electrical Engineering

Czech Technical University in Prague

© Ondřej Drbohlav & Daniel Sýkora , 2025



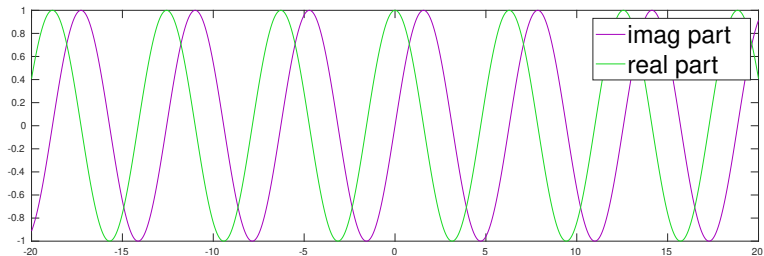
Fourier Transform

$$\text{time}\{t\} \iff \text{frequency}\{u\}$$

$$\text{forward: } F(u) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i u t} dt$$

$$\text{inverse: } f(t) = \int_{-\infty}^{\infty} F(u) e^{+2\pi i u t} du$$

$$\text{basis functions: } e^{2\pi i u t} = \cos(2\pi u t) + i \sin(2\pi u t)$$



Fourier Transform

$$\textcolor{red}{time}\{t\} \iff \textcolor{blue}{frequency}\{u\}$$

$$\text{forward: } \textcolor{blue}{F}(u) = \int_{-\infty}^{\infty} \textcolor{red}{f}(t) e^{-2\pi i u t} dt$$

$$\text{inverse: } \textcolor{red}{f}(t) = \int_{-\infty}^{\infty} \textcolor{blue}{F}(u) e^{+2\pi i u t} du$$

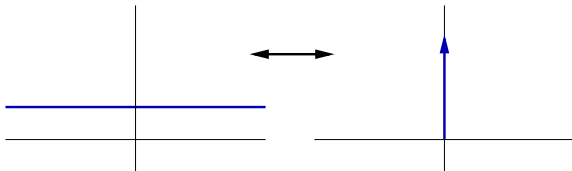
$$\text{basis functions: } e^{2\pi i u t} = \cos(2\pi u t) + i \sin(2\pi u t)$$

$$\text{amplitude: } |\textcolor{blue}{F}(u)| = \sqrt{\text{Re}(\textcolor{blue}{F}(u))^2 + \text{Im}(\textcolor{blue}{F}(u))^2}$$

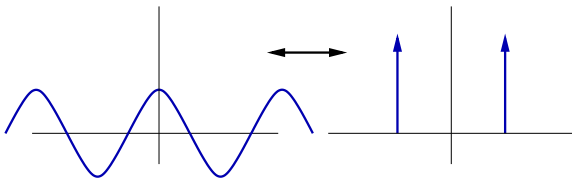
$$\text{phase: } \Phi(\textcolor{blue}{F}(u)) = \tan^{-1} \left(\frac{\text{Im}(\textcolor{blue}{F}(u))}{\text{Re}(\textcolor{blue}{F}(u))} \right)$$

Fourier Transform

$$1 \iff \delta(u)$$

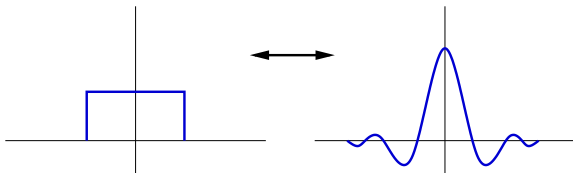


$$\cos(2\pi kx) \iff \frac{1}{2}(\delta(u+k) + \delta(u-k))$$

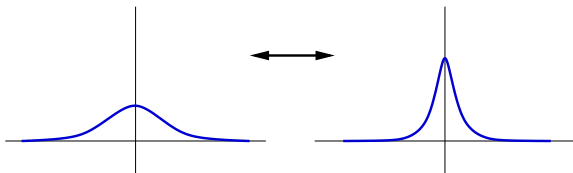


Fourier Transform

$$\mathbf{1}(x+k) - \mathbf{1}(x-k) \iff \frac{1}{\pi u} \sin(2\pi ku)$$



$$\exp(-kx^2) \iff \sqrt{\frac{\pi}{k}} \exp\left(-\frac{\pi^2}{k} u^2\right)$$



Fourier Transform — Properties

► Linearity:

$$a \cdot f(x) + b \cdot f(x) \iff a \cdot F(u) + b \cdot F(u)$$

► Parseval's theorem:

$$\int_{-\infty}^{\infty} |f(x)|^2 dx = \int_{-\infty}^{\infty} |F(u)|^2 du$$

► Convolution theorem:

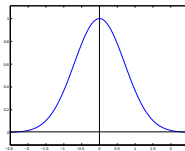
$$\int_{-\infty}^{\infty} f(x) \cdot g(t-x) dx \iff F(u) \cdot G(u)$$

► Shift theorem:

$$\int_{-\infty}^{\infty} f(x-a) e^{-2\pi i u x} dx = F(u) e^{-2\pi i u a}$$

Fourier Transform — Derivatives

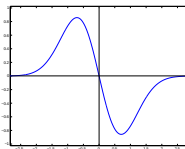
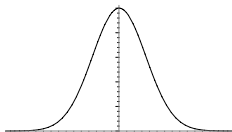
time



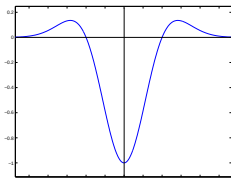
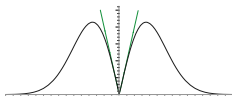
derivatives

$$\text{0th: } f(x) \iff F(u)$$

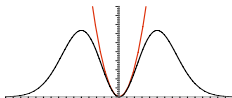
frequency



$$\text{1st: } f'(x) \iff 2\pi i u \cdot F(u)$$



$$\text{2nd: } f''(x) \iff (2\pi i u)^2 \cdot F(u)$$



Fourier Transform (in 2D)

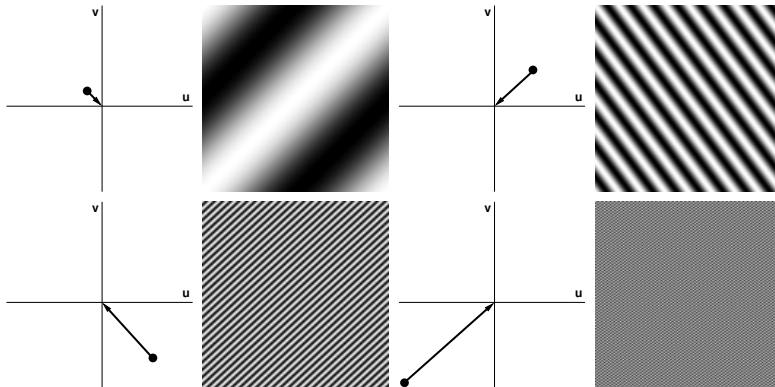
2D position $\{x, y\}$ $\iff$ *2D frequency coords* $\{u, v\}$
(or in polar coordinates: *frequency*
(scale of waves) and *orientation* (direction of waves))

forward:
$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i(ux+vy)} dx dy$$

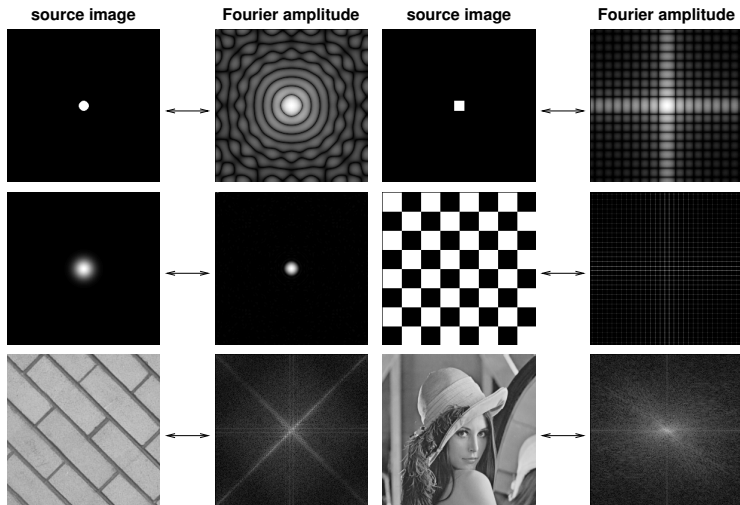
inverse:
$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{+2\pi i(ux+vy)} du dv$$

Fourier Transform (in 2D) — Basis Functions

Basis functions (only real part shown):



Fourier Transform (in 2D)



Discrete Fourier Transform (DFT)

Discrete Fourier Transform (DFT):

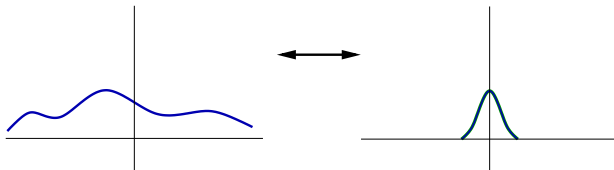
$$\text{forward: } F[u, v] = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f[x, y] e^{-2\pi i(ux+vy)/N}$$

$$\text{inverse: } f[x, y] = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F[u, v] e^{+2\pi i(ux+vy)/N}$$

Sampling Theorem

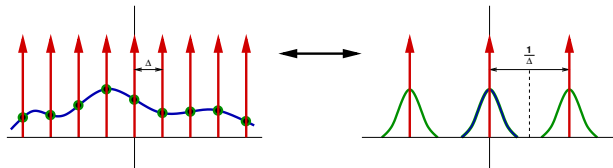
Setting:

Band-limited signal, $f_{max} \leq \frac{1}{2\Delta}$ (Nyquist frequency)



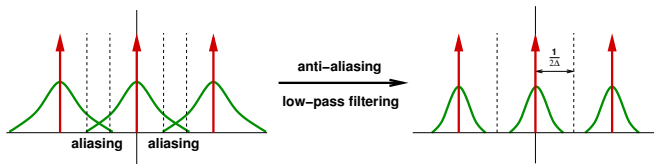
Sampling Theorem

Nyquist frequency: $f_{max} \leq \frac{1}{2\Delta}$.



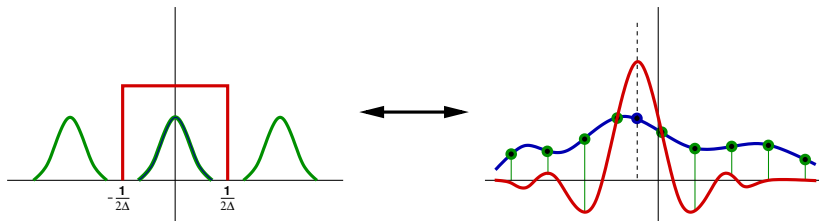
$$s(x) = \sum_k \delta(x - k\Delta) \iff S(u) = \sum_k \delta\left(u - \frac{k}{\Delta}\right)$$

$$d(x) = f(x) \cdot s(x) \iff D(u) = (F * S)(u)$$



Sampling Theorem — Signal Reconstruction

Signal reconstruction:



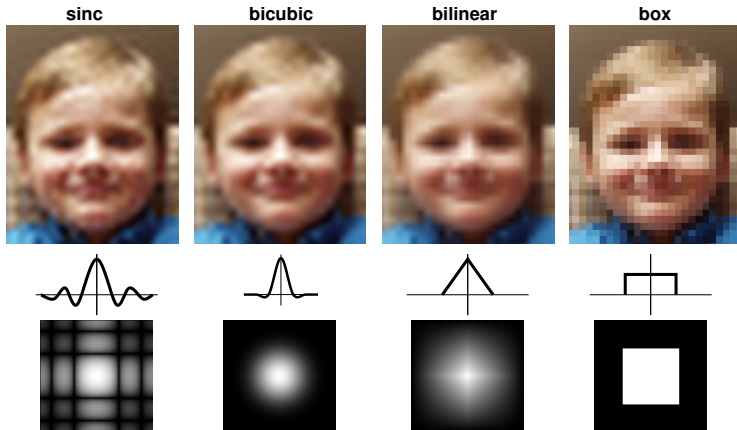
$$B(u) = \mathbf{1}\left(u + \frac{1}{2\Delta}\right) - \mathbf{1}\left(u - \frac{1}{2\Delta}\right) \iff b(x) = \frac{1}{\pi x} \sin\left(\frac{\pi x}{\Delta}\right)$$

$$F(u) = G(u) \cdot B(u) \iff f(x) = (g * b)(x)$$

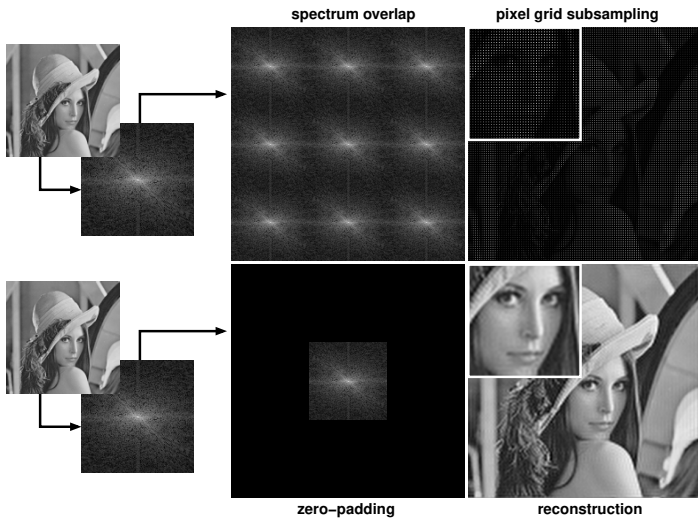
Image Resampling

Problem: convolution with **sinc** $\Rightarrow$ time consuming, ringing artifacts.

Sinc approximations with narrow support (bicubic, bilinear, box):



Example



Aliasing Example

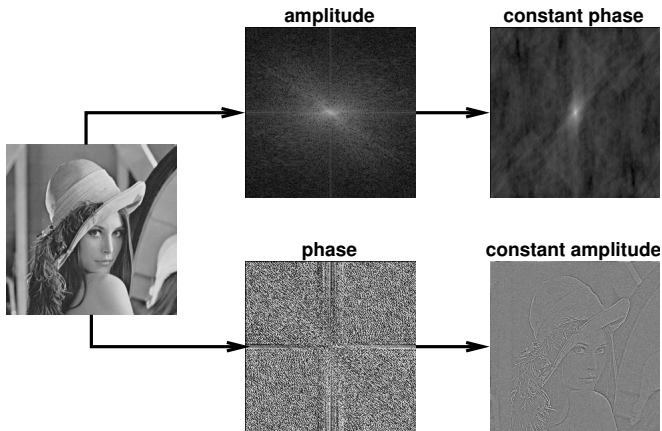


Fourier Transform — Amplitude/Phase

Edges with orientation $\arctan(v/u)$ and frequency $\sqrt{u^2 + v^2}$:

Amplitude $\Rightarrow$ intensity

Phase $\Rightarrow$ “location”

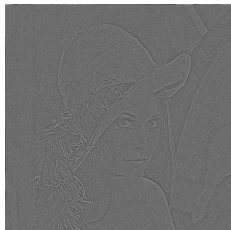


Keeping only phase (and synthesizing amplitude)

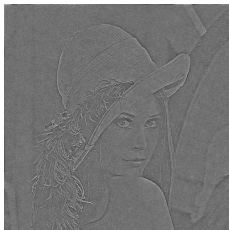
Let us keep the original phase only, and replace the amplitude by A

$$= f^{-\beta}, f = \sqrt{u^2 + v^2}:$$

$$\beta = 0$$



$$\beta = 0.4$$



$$\beta = 0.8$$



$$\beta = 1$$



$$\beta = 1.5$$



$$\beta = 2$$



Fast Fourier Transform (FFT)

Computation complexity of the naive implementation is $\mathcal{O}(N^4)$.
Can we do it faster?

Computing DFT using Fast Fourier Transform (FFT):

separability:
$$F[u, v] = \frac{1}{N} \sum_{x=0}^{N-1} e^{-2\pi i u x / N} \left(\sum_{y=0}^{N-1} f[x, y] e^{-2\pi i v y / N} \right)$$

recursion:
$$F[u] = F_{\text{even}}[u] + F_{\text{odd}}[u] e^{-2\pi i u / N},$$
$$F[u + N/2] = F_{\text{even}}[u] - F_{\text{odd}}[u] e^{-2\pi i u / N}.$$

Computation complexity: $\mathcal{O}(N^2 \log N)$.