1. Filtering in the Frequency Domain, highpass
2. Filtering in the Frequency Domain, bandpass
3. Fourier and Laplacian filter

Filtering in the Frequency Domain (reminder)

- Basic idea: Modify *importance* of «cosines» of certain frequencies by multiplying $F(u,v)$ by a filter function H(u,v) and take IDFT!

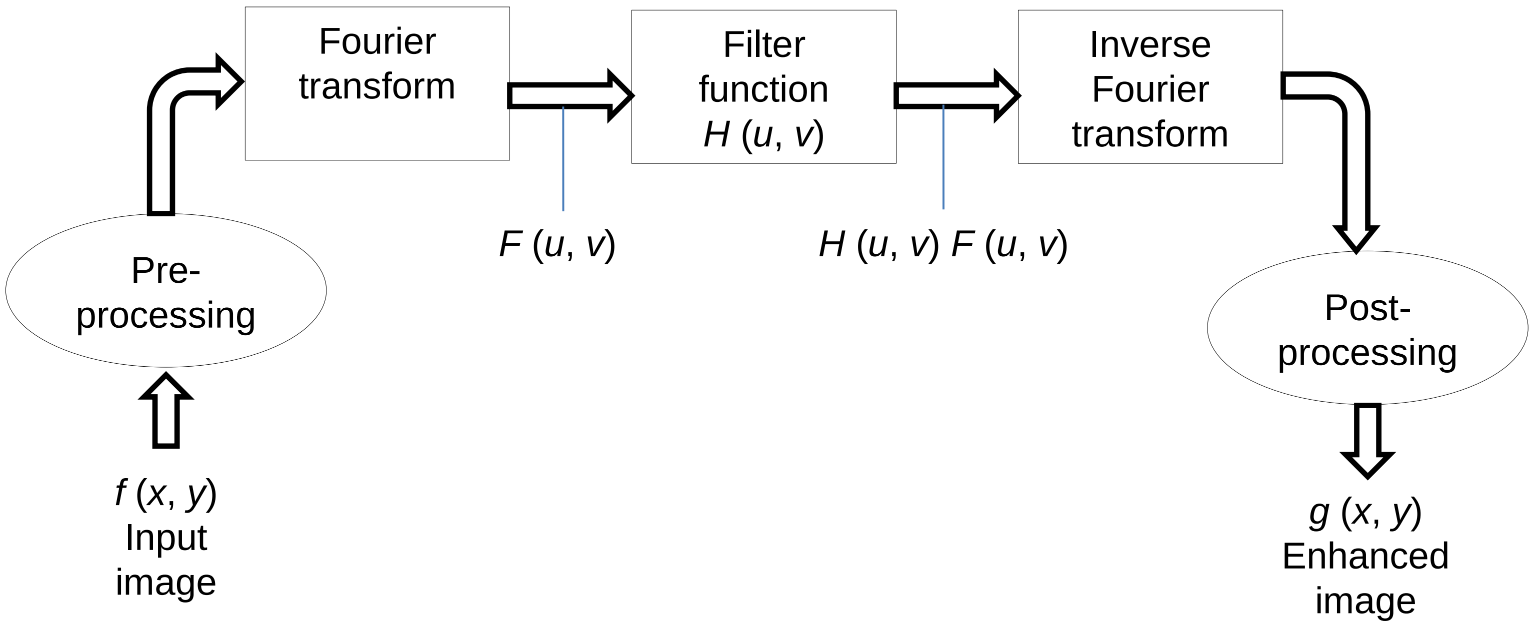


Figure: Basic steps for filtering in the frequency domain.

Filtering in the Frequency Domain

- Convolution theorem (reminder):

$$
\begin{aligned}
g(x,y) &= \mathrm{IDFT}\{\ H(u,v)F(u,v)\ \} \\
&= \mathrm{IDFT}\ \{\ H(u,v)\ \} * \mathrm{IDFT}\{\ F(u,v)\ \}
\end{aligned}
$$

- H(u,v) will often pass certain frequencies (attenuate rest)

  - Lowpass - Highpass - Selective (also Ch. 5)

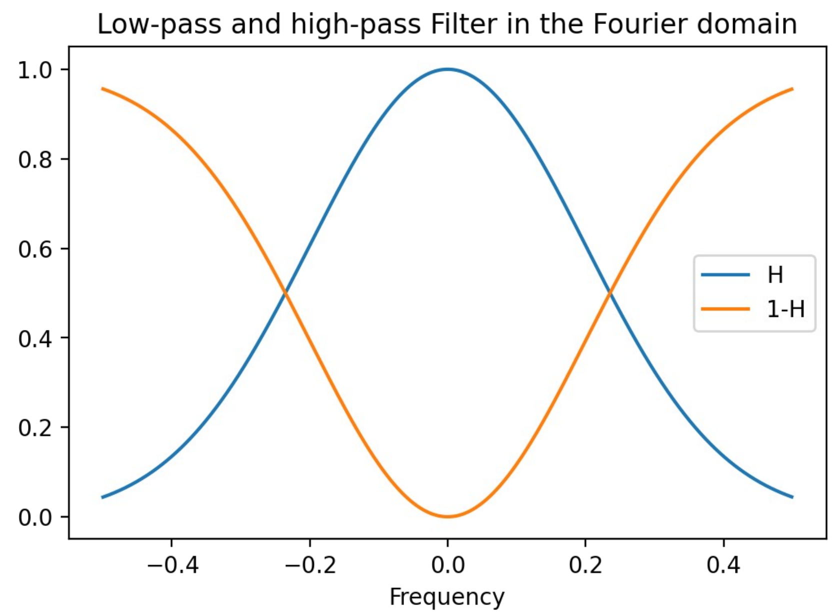
Like the convolution, filtering in the Fourier domain impact all the pixels of an image.

1. Highpass filtering and highpass filters

- «Opposite» of lowpass filter
- High frequencies are passed – low frequencies are «cut off» or attenuated

Example:

$$H_{HP}(u,v)=1-H_{LP}(u,v)$$



- Most important highpass filters:
  - Ideal
  - Gaussian
  - Butterworth

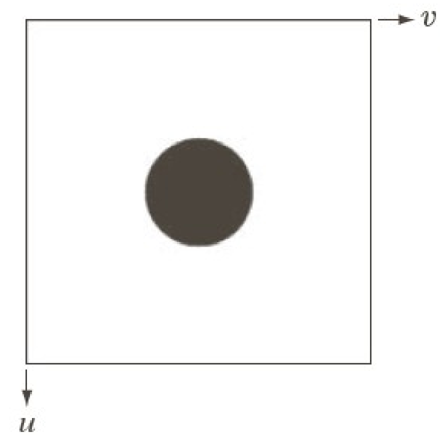


TABLE 4.5
Highpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) > D_0 \\ 0 & \text{if } D(u,v) < D_0 \end{cases}$	$H(u,v) = \dfrac{1}{1 + [D_0/D(u,v)]^{2n}}$	$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$

We can have a polynomial

We can tune the variance

High pass filtering in the Frequency Domain

Ideal

Butterworth

Gaussian

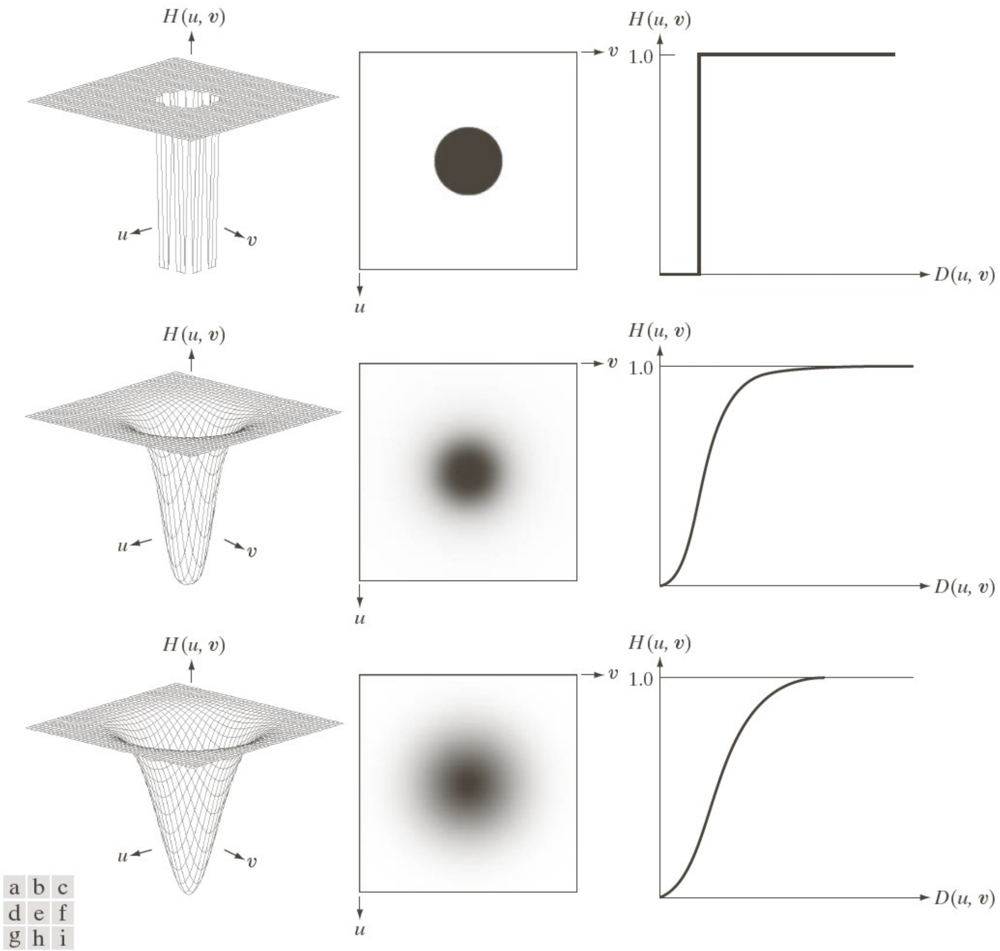
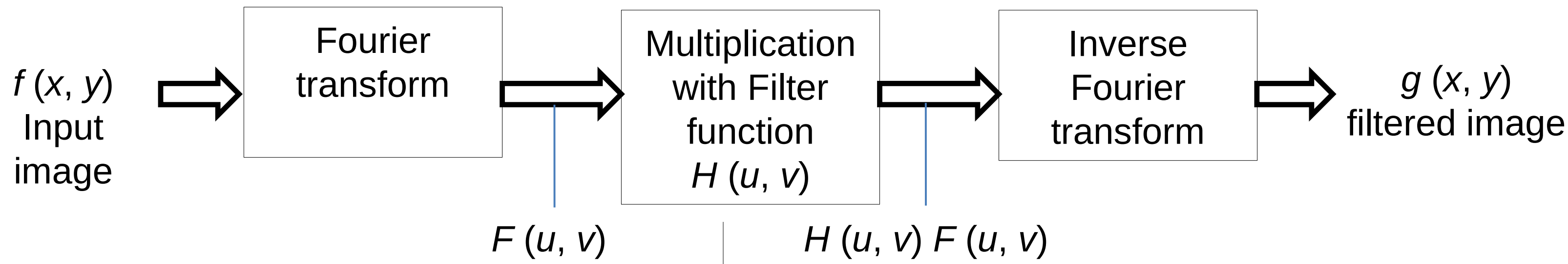
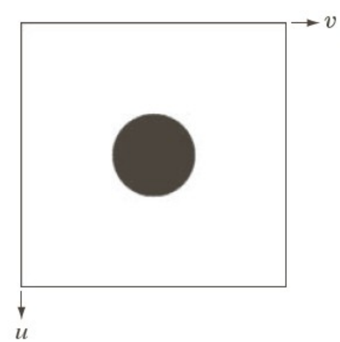


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

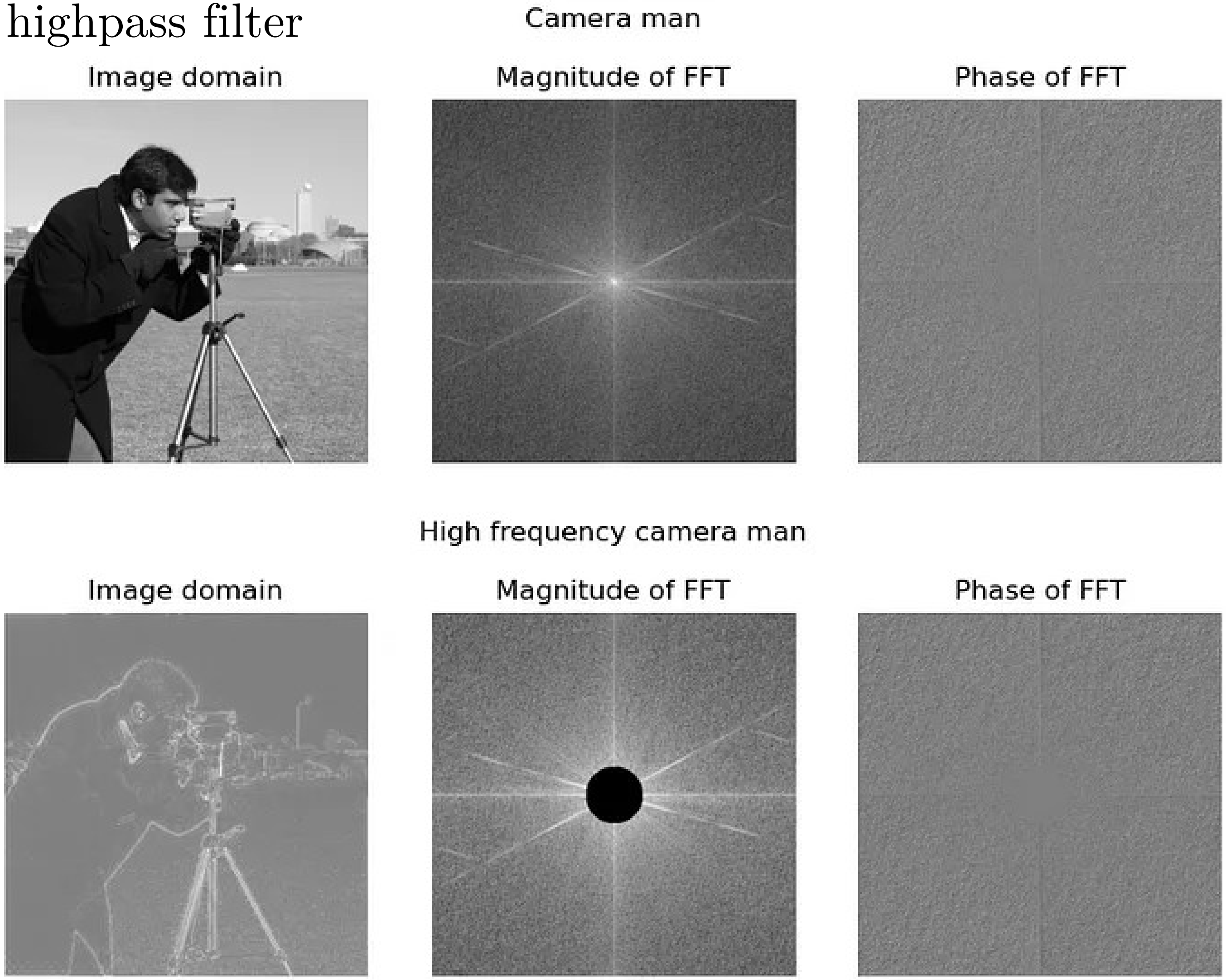
- Steps for filtering in the Fourier domain



Example of filter H(u,v):
(high-pass)



Effect of highpass filter



After the highpass filtering: only edges are visible

- Ideal highpass filter (IHPF)

Ringing effects!

a b c

FIGURE 4.54 Results of highpass filtering the image in Fig. 4.41(a) using an IHPF with $D_0 = 30$, 60, and 160.

Problem:
Artefacts due to a non-smooth filter (discontinuity in the function H)

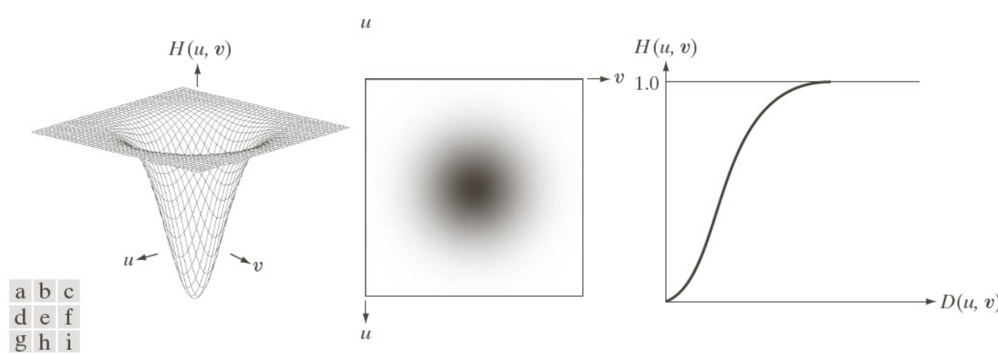
- Gaussian highpass filter (GHPF)

No ringing!

a b c

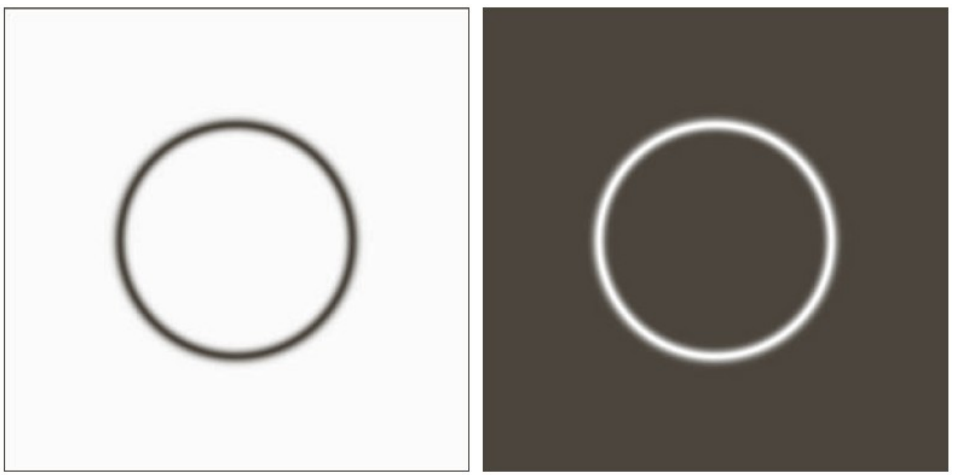
FIGURE 4.56 Results of highpass filtering the image in Fig. 4.41(a) using a GHPF with $D_0 = 30$, 60, and 160, corresponding to the circles in Fig. 4.41(b). Compare with Figs. 4.54 and 4.55.

Filter shape:



2. Selective or bandpass filters (more in Ch. 5)

- Reject or pass frequencies in a band



a b

FIGURE 4.63 (a) Bandreject Gaussian filter. (b) Corresponding bandpass filter. The thin black border in (a) was added for clarity; it is not part of the data.

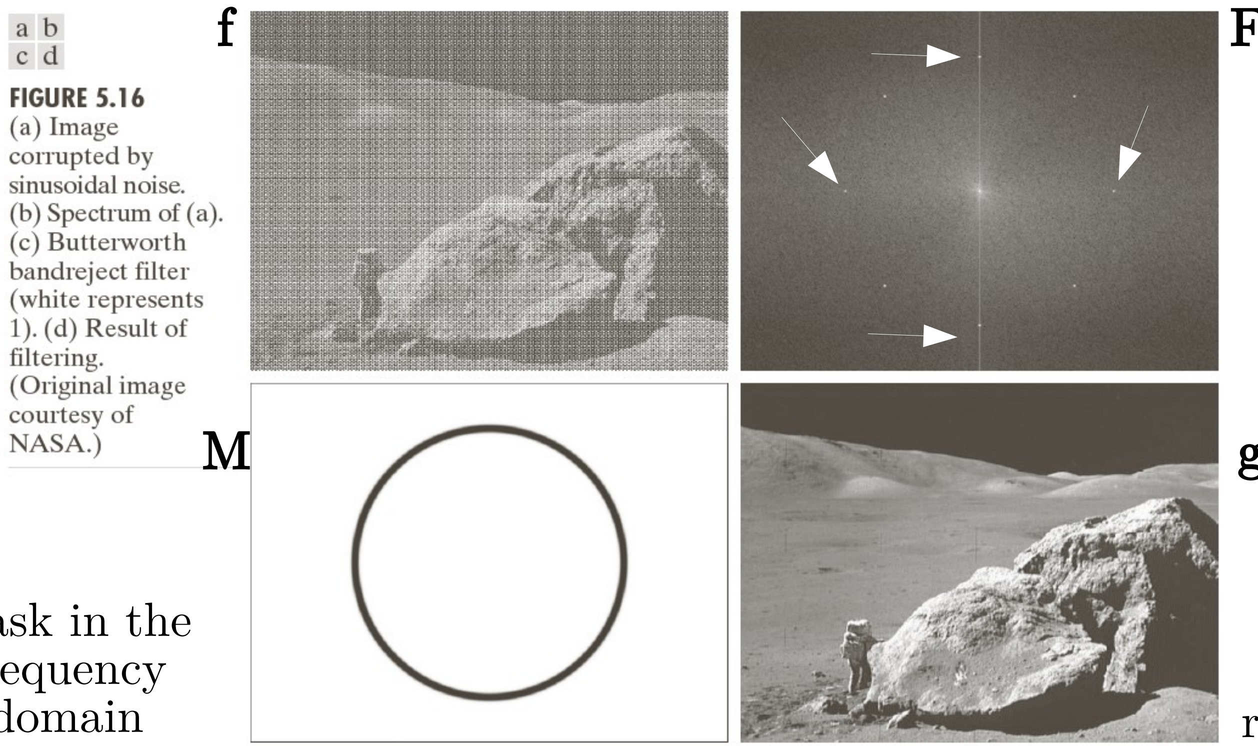


FIGURE 5.16 (a) Image corrupted by sinusoidal noise. (b) Spectrum of (a). (c) Butterworth bandreject filter (white represents 1). (d) Result of filtering. (Original image courtesy of NASA.)

Mask in the frequency domain

Image reconstructed

$$\mathbf{f} \longrightarrow \boxed{\text{DFT}} \longrightarrow \mathbf{F} \longrightarrow \boxed{\begin{array}{c}\text{Multiply by mask M} \\ \mathbf{G} = \mathbf{FM}\end{array}} \longrightarrow \mathbf{G} \longrightarrow \boxed{\text{IDFT}} \longrightarrow \mathbf{g}$$

Note: the part masked is larger than the noisy spots, but it does not impact much the image

Filtering in the Frequency Domain, summary

- We may filter images in the frequency domain by multiplication and inverse DFT
- Achieve same results as the convolution techniques of Chapter 3
- *Useful in analysis*, and interpretation.
- More efficient for removing periodic noise

Recent research in machine learning related to Fourier:

Fourier neural operators for parametric differential equations:
https://arxiv.org/abs/2010.08895 and explanation:
https://www.youtube.com/watch?v=IaS72aHrJKE

- In our machine learning group:
- https://arxiv.org/abs/2201.07544
- https://www.sciencedirect.com/science/article/pii/S1631070519301094

3. The Laplacian comes back

- Applying the Laplacian filter keeps only the sharp edges of the image: the Laplacian filter is an highpass filter ??

Reminder: the Laplacian is the sum of the second derivatives
For example in 2d:

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

And in 1d discrete:

$$\frac{\partial^2 f}{\partial x^2}(x) = f(x+1) + f(x-1) - 2f(x)$$

The derivative operation is linear: D(af+bg)=aDf+bDg
The Laplacian is a linear operator, it is a matrix in the discrete domain

The Laplacian on a periodic 1d domain (ring)

$$\Delta_p = \begin{pmatrix} 2 & -1 & & & & -1 \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ -1 & & & & -1 & 2 \end{pmatrix}$$

$$\frac{\partial^2 f}{\partial x^2}(x) = -f(x+1) - f(x-1) + 2f(x)$$

- The Laplacian matrix is symmetric with real entries,
- *Do you see the relationship with the convolution?*

We can diagonalize this matrix to get the eigenvalues and eigenvectors. What are they?

Fourier Transformation & Laplacian

Fourier transform

$$F(k) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi kx} dx$$

Fourier modes $\psi_k(x) = e^{-2i\pi kx}$

Laplacian of the Fourier modes :

$$\Delta e^{-2i\pi kx} = -4\pi^2 k^2 e^{-2i\pi kx}$$

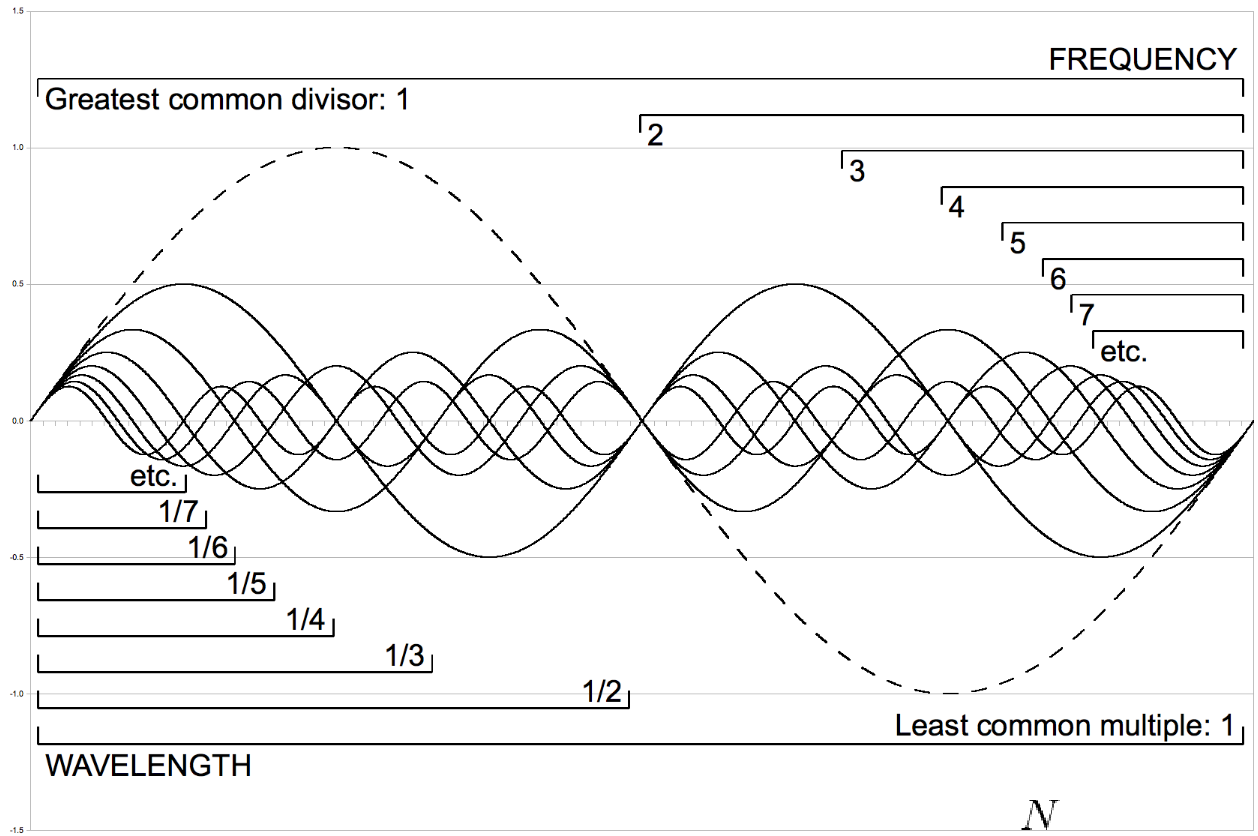
$$\Delta\psi_k(x) = \lambda_k \psi_k(x) \qquad \lambda_k = -4\pi^2 k^2$$

Eigenvectors of the Laplacian !

Note : these are «generalized» eigenvectors

The Laplacian eigenvectors

• On a continuous line or ring: eigenvectors of the Laplacian are Fourier modes $e^{2i\pi fx} = \cos(2\pi fx) + i\sin(2\pi fx)$



Fourier modes are orthogonal (exercise)

$$(\psi_i, \psi_k) = \sum_{n=0}^{N} \psi_i^*(n)\psi_k(n) = \delta_{ik}$$

Fourier and derivative

- Property of continuous FT:

$$
\begin{array}{ccc}
f(x,y) & \text{<--->} & F(u,v) \\
\partial^m f/\partial x^m & \text{<--->} & (i2\pi u)^m F(u,v) \\
\partial^m f/\partial y^m & \text{<--->} & (i2\pi v)^m F(u,v)
\end{array}
$$

Obtained using integration by parts:

$$\int_{-\infty}^{\infty} \Delta f(x)\psi_k(x)dx = -\int_{-\infty}^{\infty} f(x)\Delta\psi_k(x)dx$$

We have

$$\int_{-\infty}^{\infty} \Delta f(x)\psi_k(x)dx = \int_{-\infty}^{\infty} 4\pi^2 k^2 f(x)\psi_k(x)dx = 4\pi^2 k^2 F(k)$$

It is a highpass filter! (do you see why?)

Laplacian sharpening: convolution -> (all-pass + high-pass) filter:

$$g(x,y) = f(x,y) + c\Delta f(x,y)$$

Fourier domain: $G(u,v) = \left(1 + c4\pi^2(u^2+v^2)\right) F(u,v)$

a b

FIGURE 4.58 (a) Original, blurry image. (b) Image enhanced using the Laplacian in the frequency domain. Compare with Fig. 3.38(e).

-> Sharpening is increasing the high frequencies content

Sharpening in the discrete domain (optional)

- Discrete domain, DFT:

$$Lf[n] = f[n+1] + f[n-1] - 2f[n] = f * (\delta_1 + \delta_{-1} - 2\delta_0)[n]$$

$$DFT(\delta_s)[k] = \sum_{n=0}^{N-1} \delta(n-s) e^{2i\pi nk/N} = e^{2i\pi sk/N}$$ (n-s) mod N, periodic

$$DFT(Lf)[k] = F[k](e^{2i\pi k/N} + e^{-2i\pi k/N} - 2) = 2F[k](\cos(2\pi k/N) - 1)$$

$$DFT(-Lf)[k] = 2(1 - \cos(2\pi k/N))F[k]$$

High-pass filter !