U4. Image Filtering: Frequency Domain

SJK002 Computer Vision

Master in Intelligent Sytems

UNIVERSITAT JAUME I

Index

- Introduction
- Discrete Fourier Transform (DFT):
  - FT visualization
- Fast Fourier Transform (FFT)
- Uses of FT
- Convolution Theorem
- Filtering in the frequency domain
  - Low pass filter
  - High pass filter
  - Filtering specific frequencies

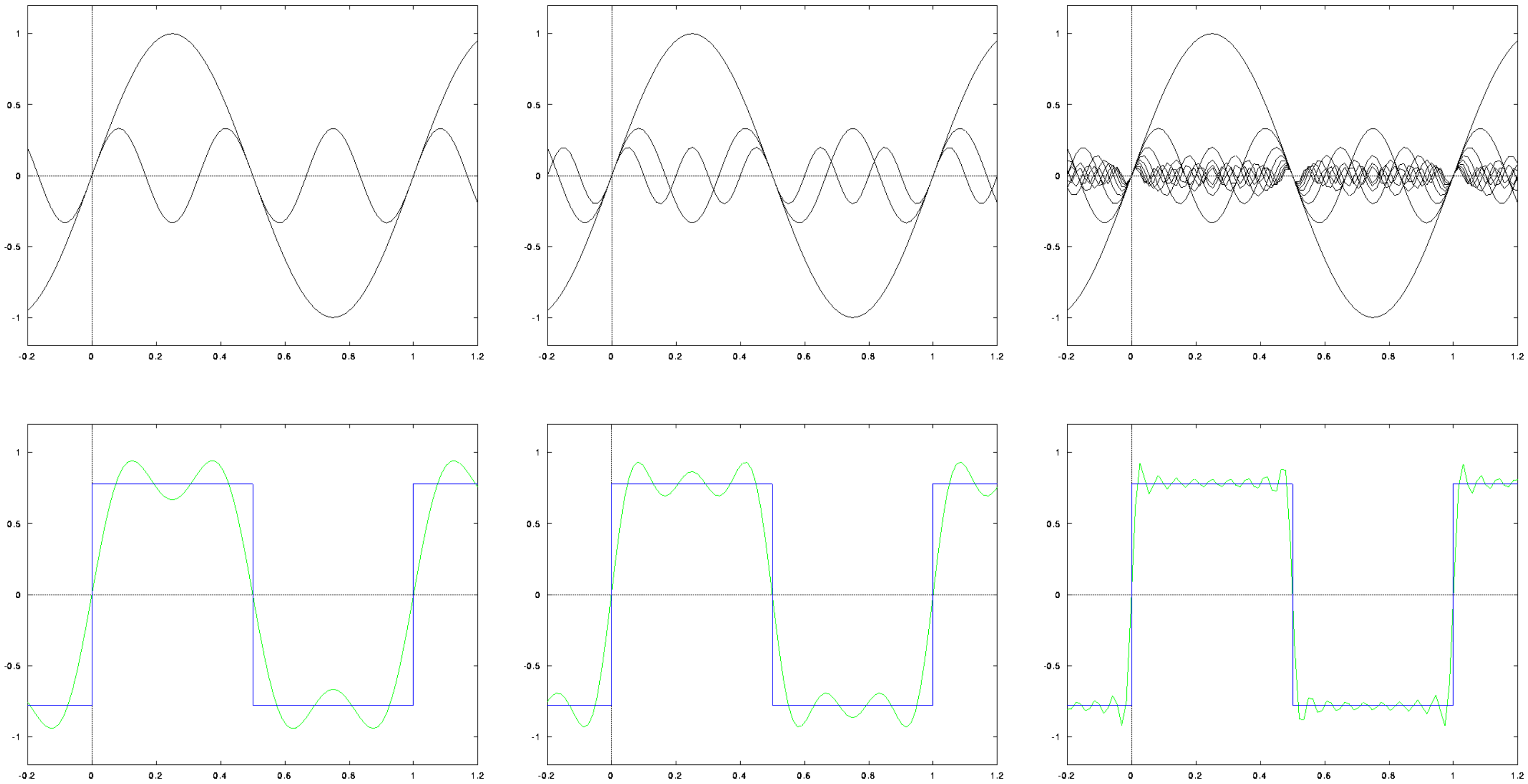
Introduction

- Same objective as filtering in the spatial domain:
  - Achieve an appearance improvement of the original image
  - Enhance certain features
  - Eliminate noise (which masks some features)

- Filtering with different points of views:
  - Spatial domain → Spatial coordinates (x,y)
  - Frequency domain → Spatial frequency (u,v)

Introduction

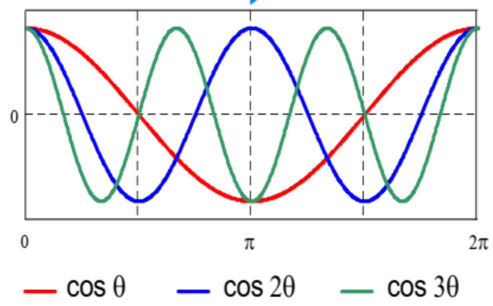
- Combining several sinusoidal functions, any signal function can be represented or approximated

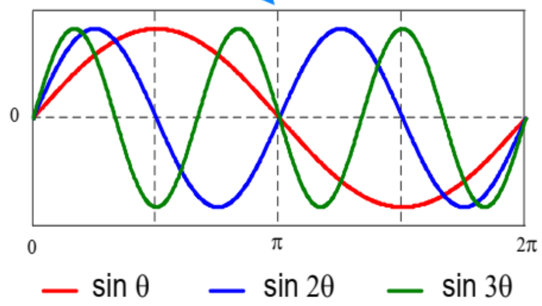


Fourier Transform

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(nx) + \sum_{n=1}^{\infty} b_n \sin(nx)$$

Development in a Fourier series





Coefficients of the Fourier series

$$a_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(nx)\,dx \quad n = 0,1,2,\ldots$$

$$b_n = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(nx)\,dx \quad n = 1,2,3,\ldots$$

$$\cos\varphi = \frac{e^{i\varphi} + e^{-i\varphi}}{2} \qquad \sin\varphi = \frac{e^{i\varphi} - e^{-i\varphi}}{2i}$$

Fourier Transform

Direct Fourier Transform

$$F(\omega) = \mathcal{F}[f](\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) \mathrm{e}^{-i\omega x}\, dx$$

Inverse Fourier Transform

$$f(x) = \mathcal{F}^{-1}[F](x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(\omega) \mathrm{e}^{i\omega x}\, d\omega$$

$$F[k] = \sum_{n=0}^{N-1} f[n] \mathrm{e}^{-2\pi i n k/N} \qquad (k = 0, 1, \ldots, N-1)$$

Discrete Fourier Transform

2D discrete Fourier Transform

$$F[k,j] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f[n,m] \mathrm{e}^{-2\pi i (kn/N + jm/M)}$$

$$(j = 0, 1, \ldots, M-1; k = 0, 1, \ldots, N-1)$$

Discrete Fourier Transform (DFT)

- Each frequency component is a complex number:
  - Real part / imaginary part

$$H(u,v) = R(u,v) + j \cdot I(u,v)$$

$$j = \sqrt{-1}$$

  - Magnitude / Phase

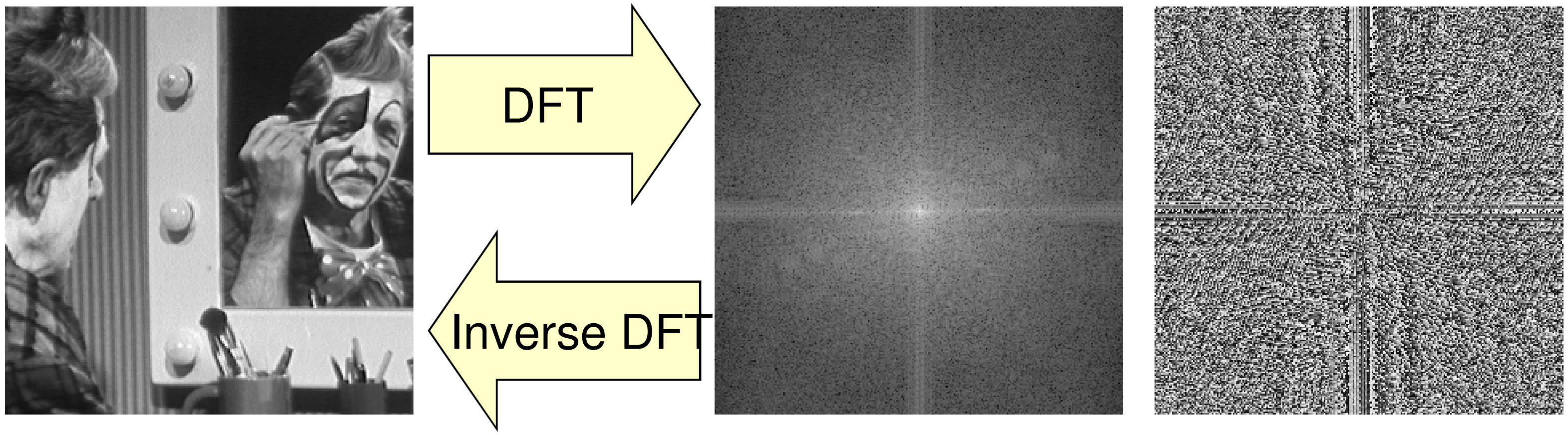
$$|H(u,v)| = \sqrt{R^2(u,v) + I^2(u,v)}$$

$$\Theta(u,v) = \tan^{-1}\left[\frac{I(u,v)}{R(u,v)}\right]$$

Discrete Fourier Transform (DFT)

- The transformation is reversible:

Original image | DFT magnitude | DFT phase

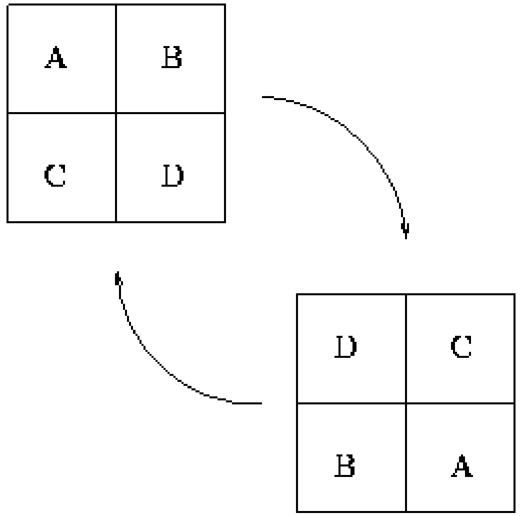


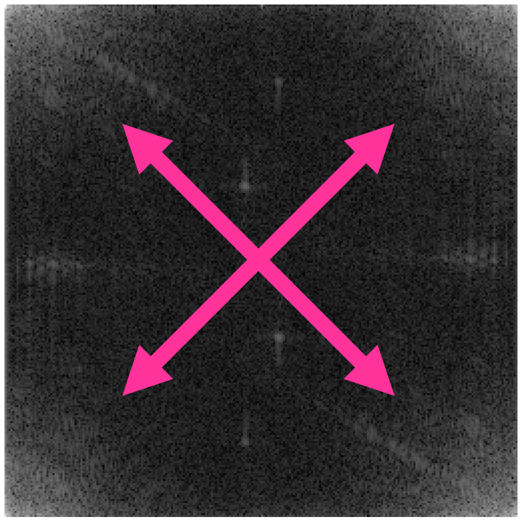
$h(x,y)$ | $|H(u,v)|$ | $\Theta(u,v)$

- If the input image is real:
  - Real part / Magnitude are symmetric
  - Imaginary part / Phase are antisymmetric
  - ➢ It is enough using half of the values
- It is a linear operation

Visualization of the DFT

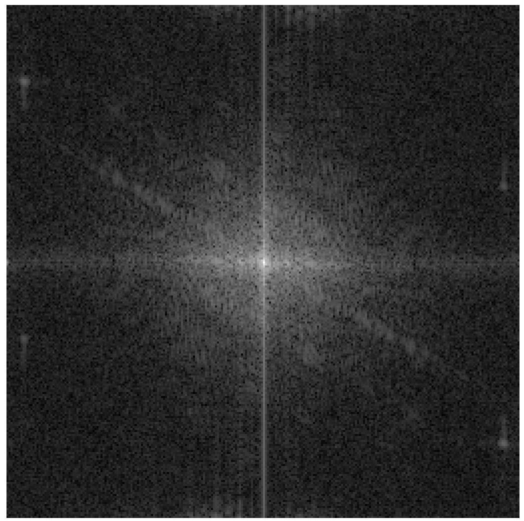
- Re-ordering values to obtain the optic DFT



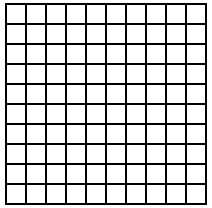


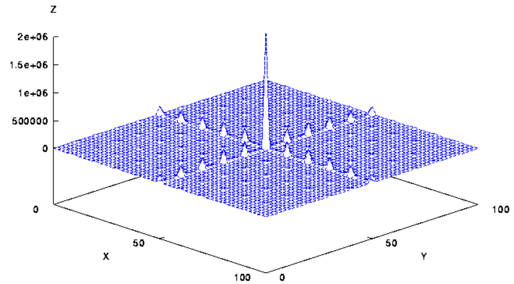
Standard DFT

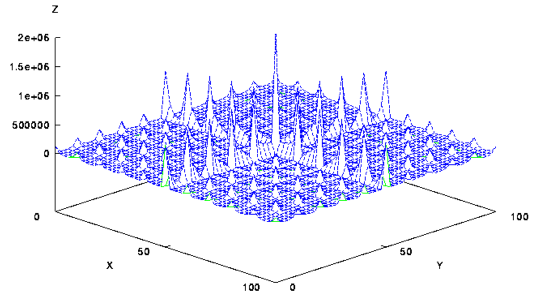
Optic DFT



Visualization of the DFT



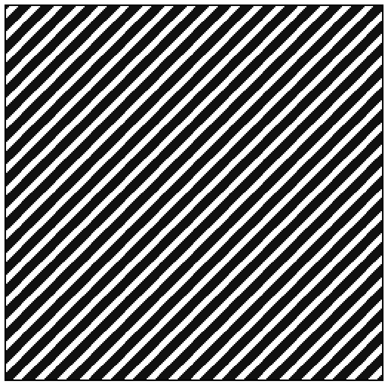


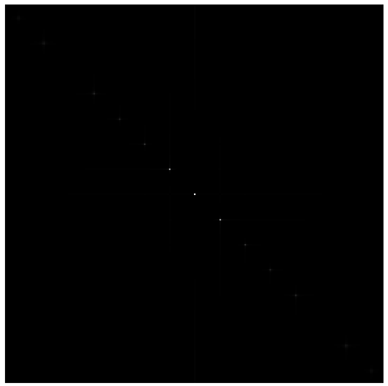


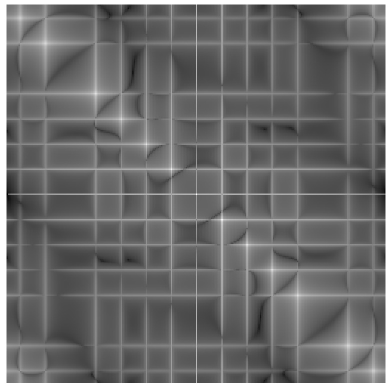
Image

DFT magnitude

DFT magnitude (Logarithmic scale)







DFT / FFT

- In 1D signals for N samples
  - DFT: $O(N^2)$
    Holds for any N value
  - FFT: $O(N \log N)$
    Only holds for N values that are power 2
    (*Zero padding*)

- In 2D:
  - The transform is separable
  - Transform first in one dimension and then in the other dimension

Convolution Theorem

- Filtering (Convolution Theorem)

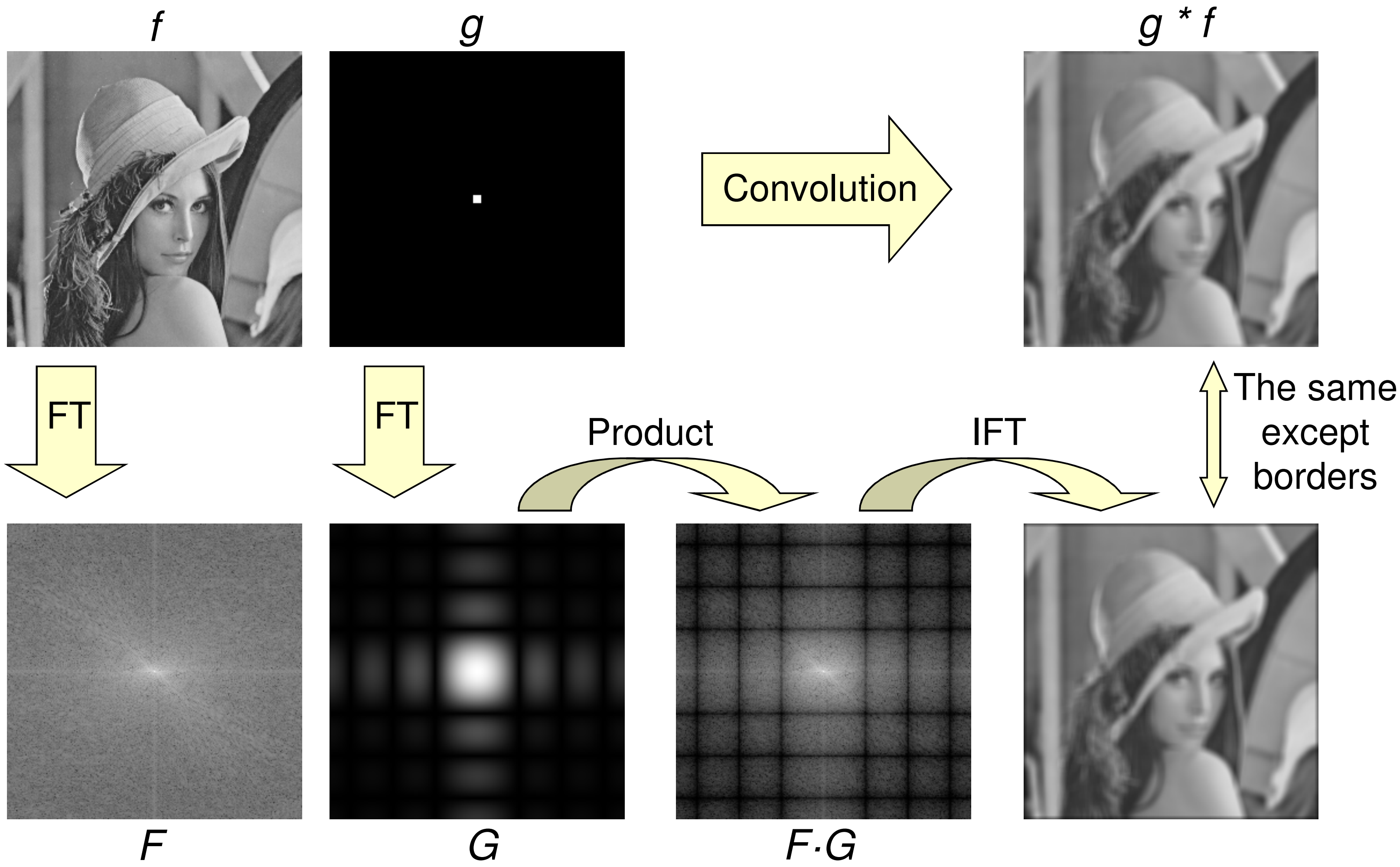
  The convolution in the spatial domain is equivalent to a product in the frequency domain:

$$f * g = h \qquad f \xrightarrow{\text{FT}} F \qquad F \cdot G = H$$

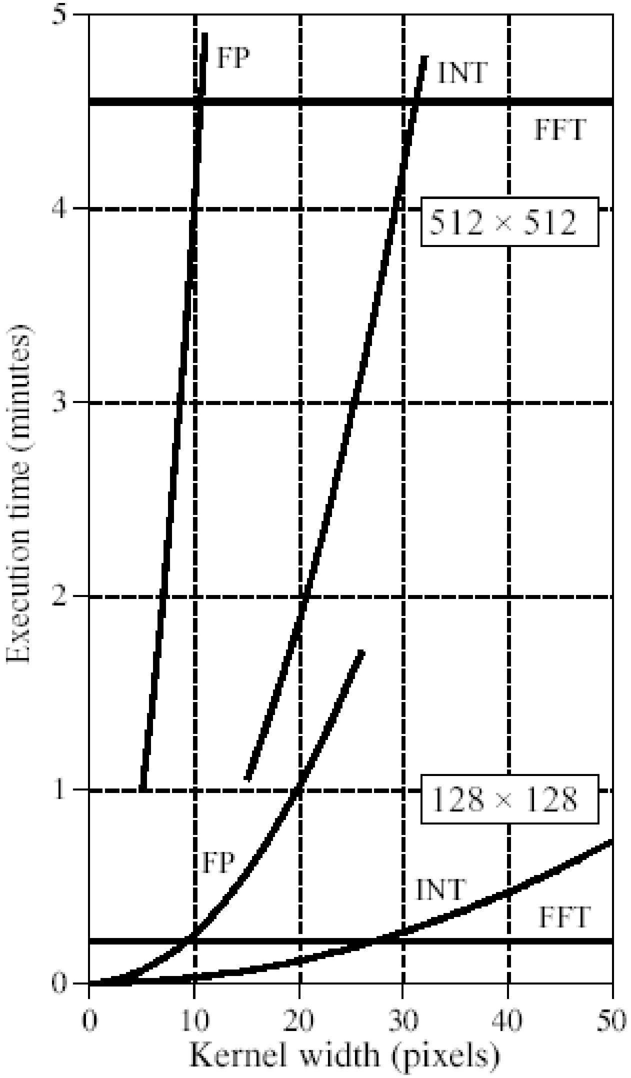
$$g \xrightarrow{\text{FT}} G$$

$$h \xrightarrow{\text{FT}} H$$

Convolution Theorem

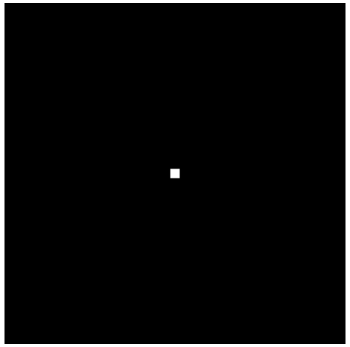


Convolution Theorem

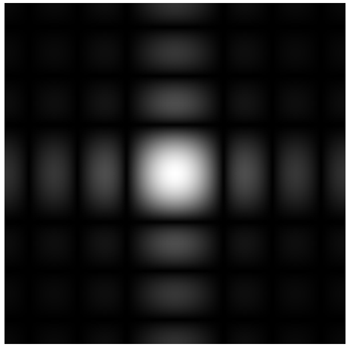


Filtering in the spatial/frequency

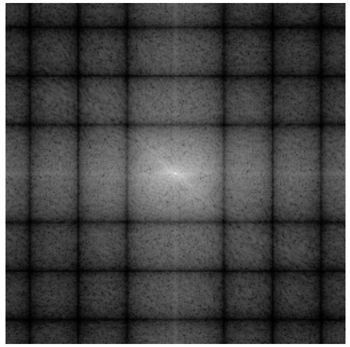
- Mean filter vs. Gaussian filter



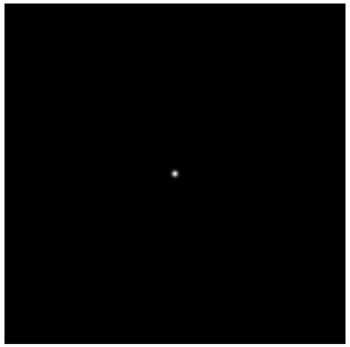
7x7 mean filter
in spatial domain



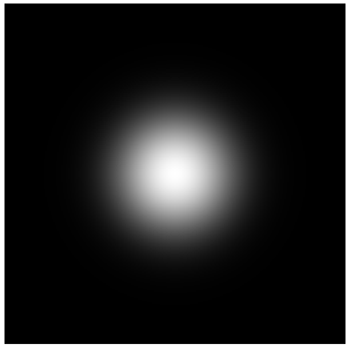
7x7 mean filter
in frequency domain



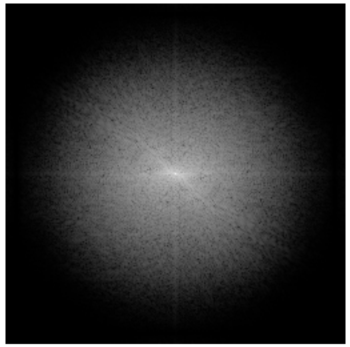
Magnitude of
filtered Lena image



Gaussian filter in
spatial domain (σ=1.5)

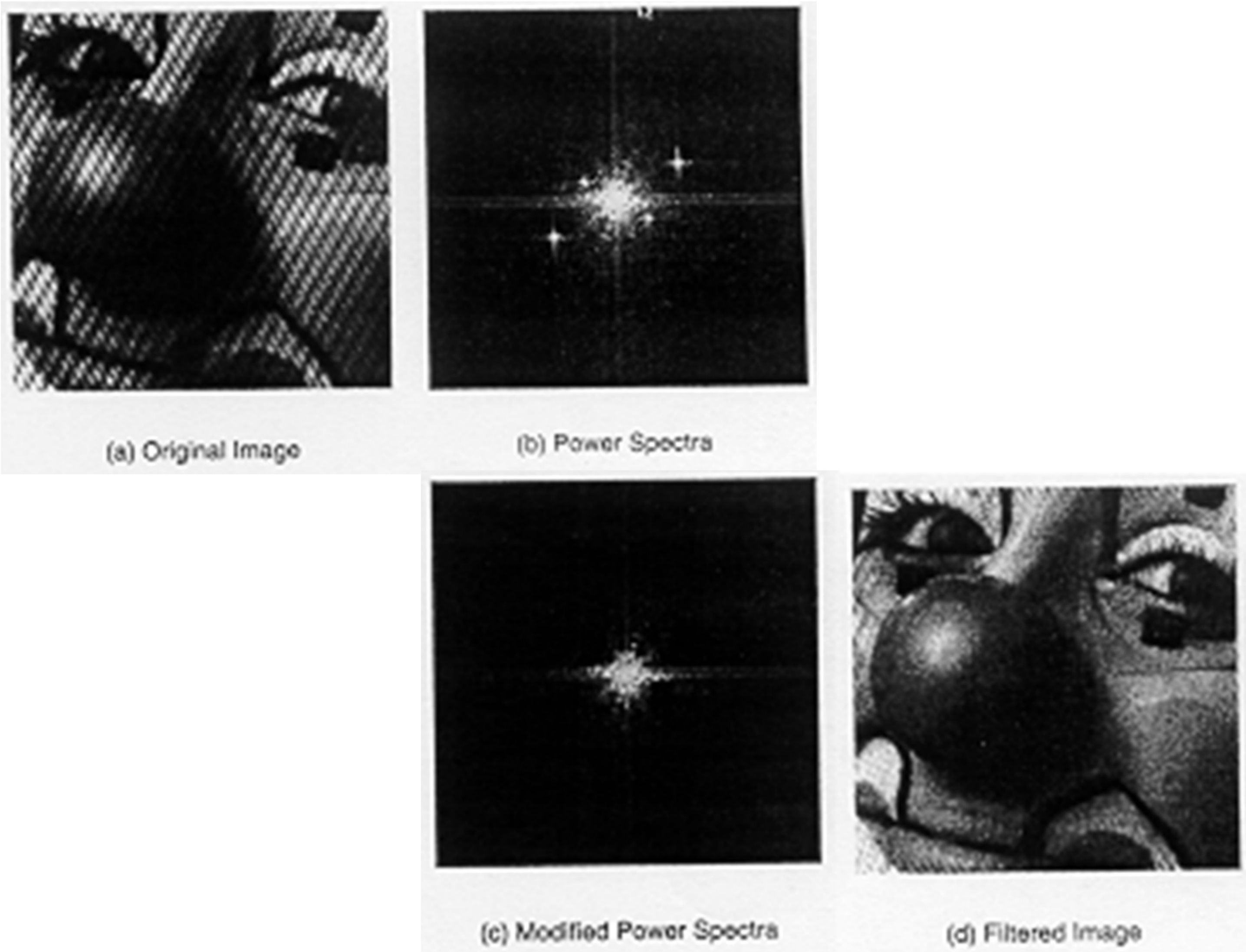


Gaussian filter in
frequency domain



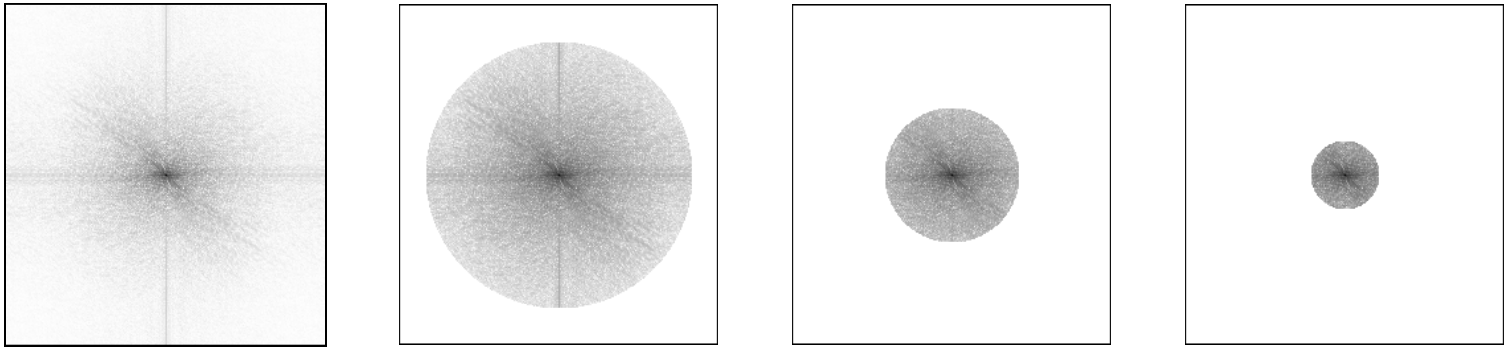
Magnitude of
filtered Lena image

Filtering in the frequency domain



Filtering in the frequency domain

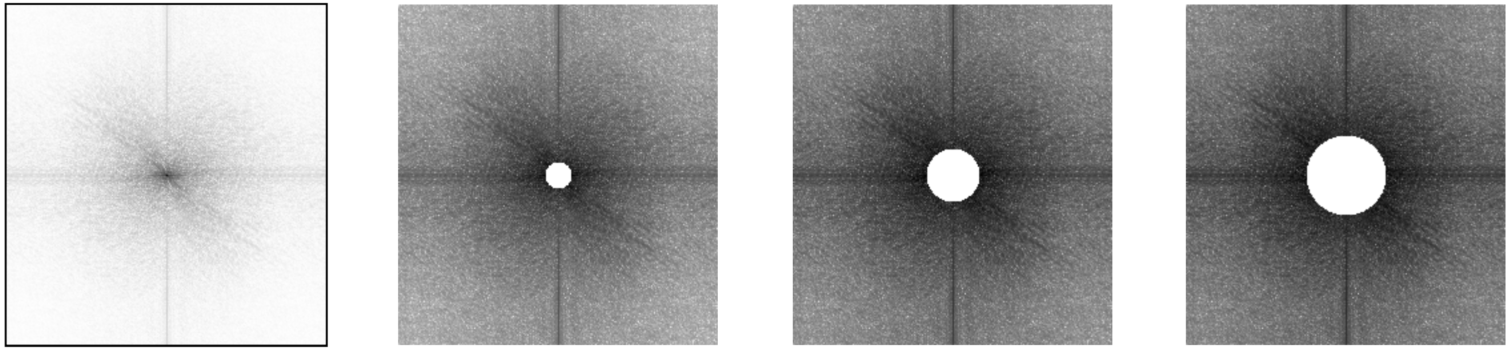
- Low pass filter



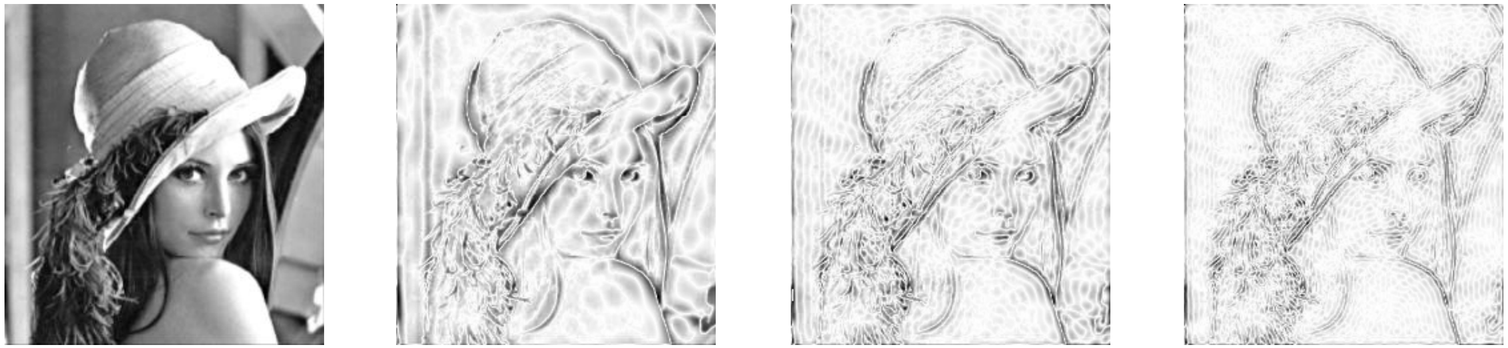
(FT in inverse gray level values)

Filtering in the frequency domain

- High pass filter



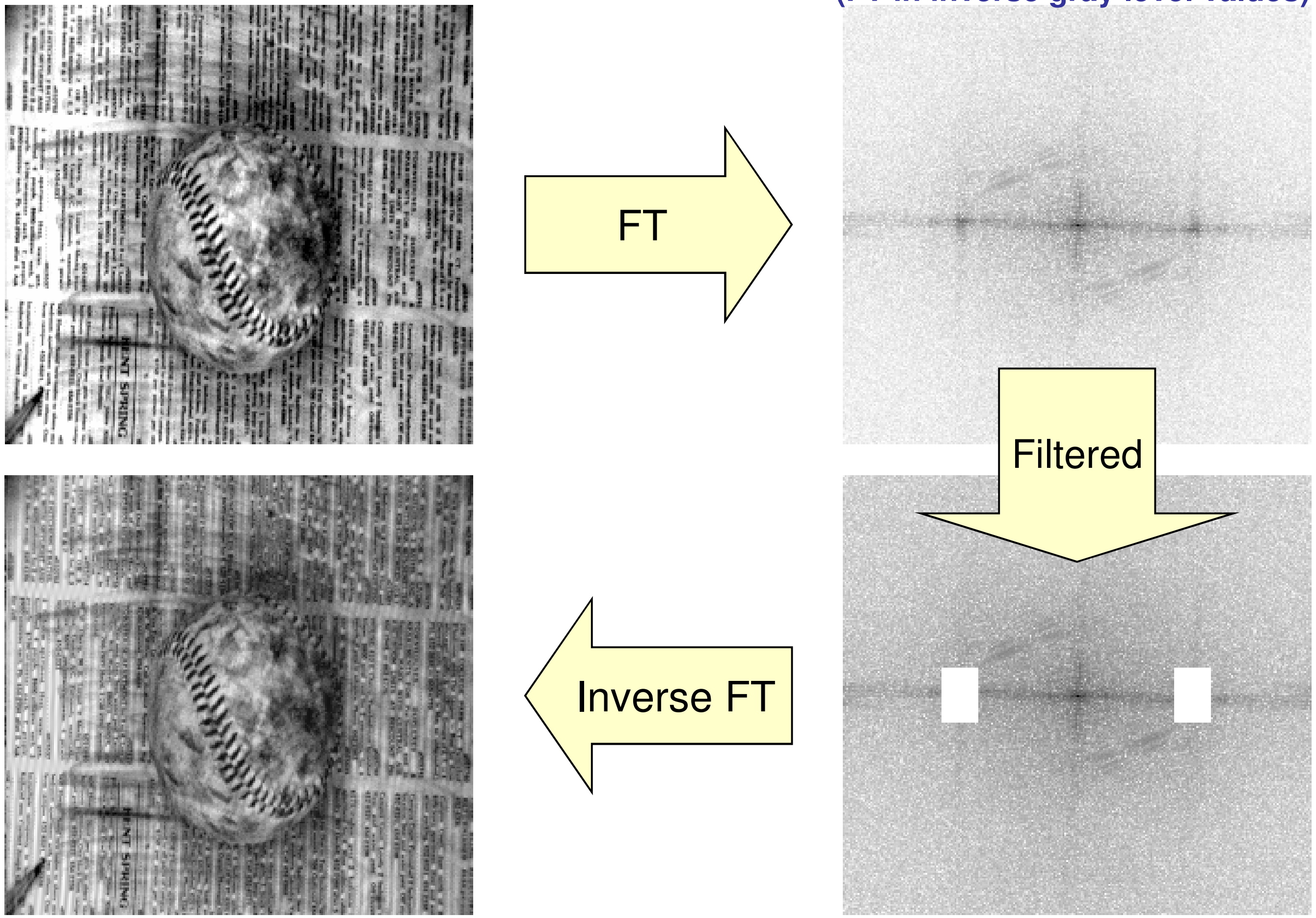
(FT in inverse gray level values)



Filtering in the frequency domain

- Filtering specific frequencies

(FT in inverse gray level values)



Bibliography

- http://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm
- Numerical Recipes in C, the art of scientific computing. Cambridge University Press. Capítulos 12 y 13. (www.nr.com)
- The scientist and engineer's guide to Digital Signal Processing, by Steven W. Smith (http://www.dspguide.com/pdfbook.htm)
- Image Processing. Dealing with texture. Maria Petrou and Pedro García-Sevilla, Wiley, 2006.