EEE-6512: Image Processing and Computer Vision

October 20, 2017
Lecture #8: Frequency Domain Processing
Damon L. Woodard, Ph.D.
Dept. of Electrical and Computer Engineering
dwoodard@ece.ufl.edu

Lecture Outline

- Preliminaries
- Fourier Transform
- Discrete Fourier Transform (DFT)
- Two-Dimensional DFT
- Frequency-Domain Filtering
- Localizing Frequencies in Time
- Discrete Wavelet Transform

Two-Dimensional DFT

Two-Dimensional DFT

- The 2D DFT is a natural extension of the 1D case:
  - Replace the single frequency k with two frequencies in the two directions, k_x and k_y, so that kx/w becomes $k_x x/w + k_y y/h$.

$$G(k_x, k_y) = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} g(x, y) e^{-j2\pi \mathbf{x}^{\mathsf{T}}\mathbf{f}} \qquad \text{(forward DFT)}$$

$$g(x, y) = \frac{1}{wh} \sum_{k_x=0}^{w-1} \sum_{k_y=0}^{h-1} G(k_x, k_y) e^{j2\pi \mathbf{x}^{\mathsf{T}}\mathbf{f}} \qquad \text{(inverse DFT)}$$

Separability

- The separability of the 2D DFT leads to a compact representation of the 2D DFT using matrix notation.

$$\mathbf{G}_{\mathcal{F}} = \mathbf{F}_h \mathbf{G} \mathbf{F}_w$$

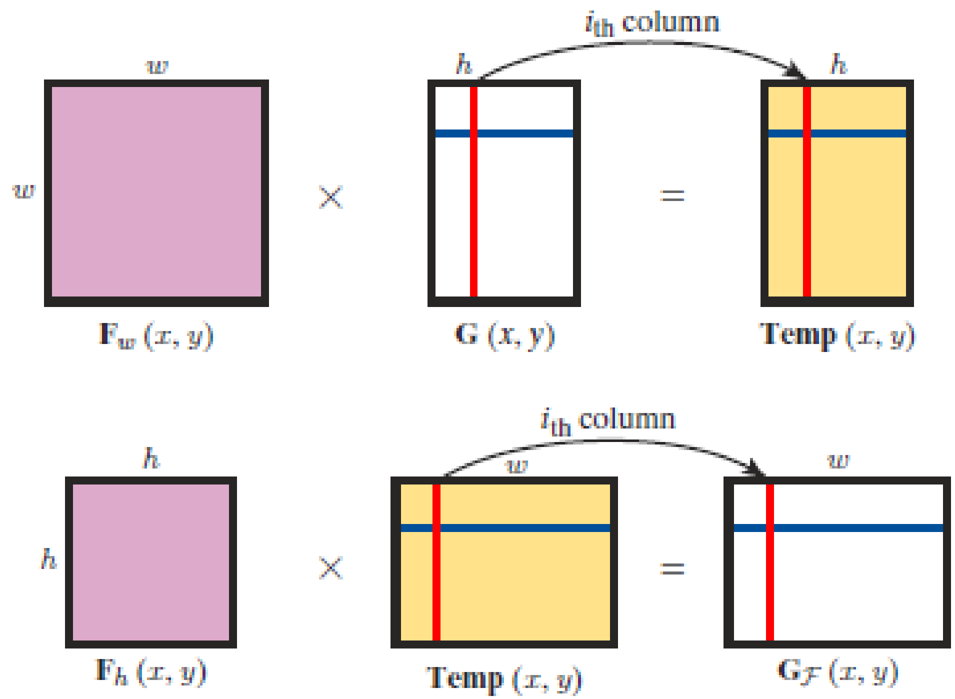


Figure 6.7 The 2D DFT as a pair of matrix multiplies, utilizing the principle of separability. The 1D DFT matrix is multiplied by the transpose of the original signal (treated as a matrix) to compute the 1D DFTs along the rows of $\mathbf{G}$ (columns of $\mathbf{G}^T$). Then this result is premultiplied by the 1D DFT matrix to compute the 1D DFTs along the columns of $\mathbf{G}$, yielding the 2D DFT $\mathbf{G}_{\mathcal{F}}$.

Projection-Slice Theorem

- The **projection** of a continuous function $g(x, y)$ of two variables onto a line at some orientation θ is the 1D function that results from integrating the function along rays perpendicular to the line.

- **Slice:** defined through a 2D continuous function $G(f_x, f_y)$ at θ as the 1D function obtained by ignoring all values except those along the line.

Projection-Slice Theorem (cont'd)

The **projection-slice theorem**:

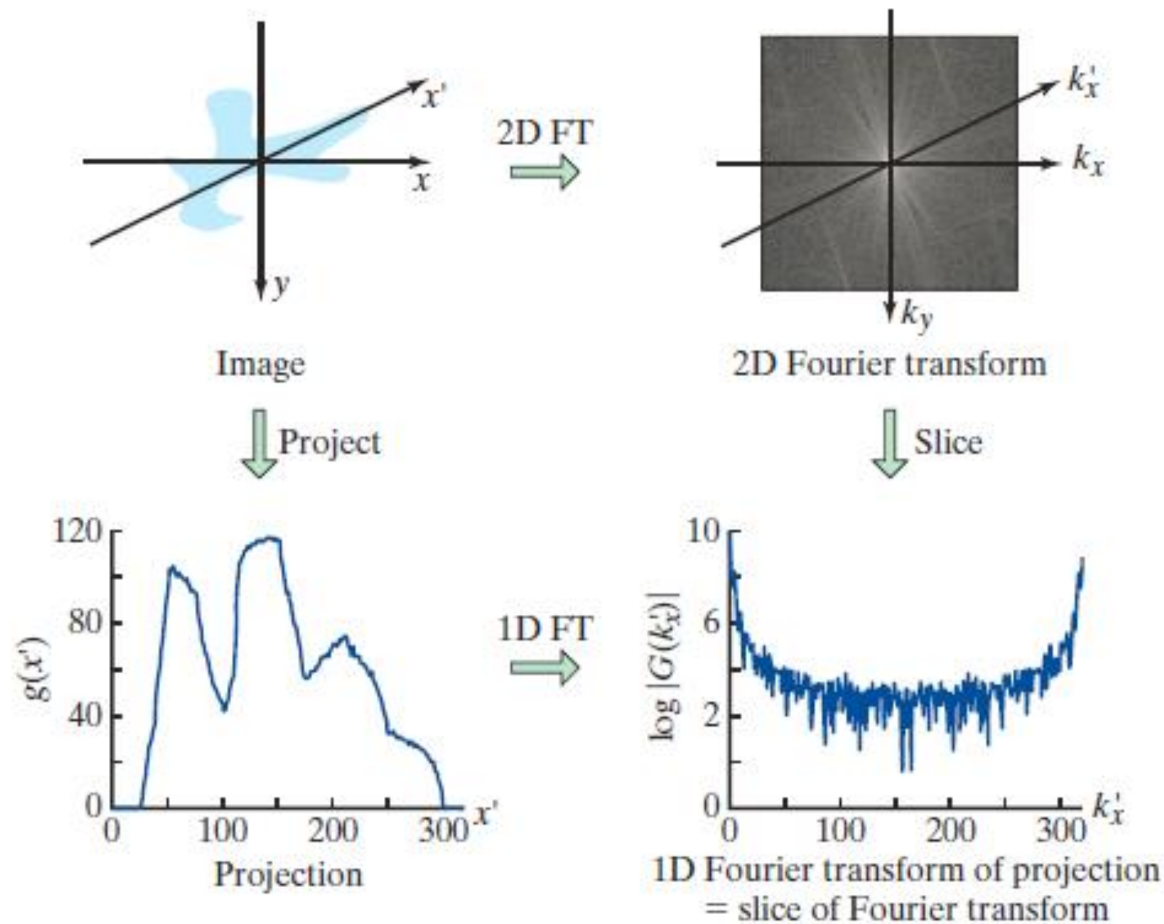
- Also known as the **Fourier slice theorem**
- Says that the Fourier transform of the projection of *g* onto a line through the origin is the same as the 1D slice of *G* at the same orientation, where

$$G = \mathcal{F}\{g\}$$

$$\begin{aligned} G(f_x, 0) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) e^{-j2\pi f_x x} \, dx \, dy \\ &= \int_{-\infty}^{\infty} \underbrace{\left[\int_{-\infty}^{\infty} g(x, y) dy \right]}_{g_p(x)} e^{-j2\pi f_x x} \, d \\ &= \mathcal{F}\{g_p(x)\} \end{aligned}$$

Projection-Slice Theorem (cont'd)

Figure 6.8 Projection-slice theorem.



Displaying the 2D DFT

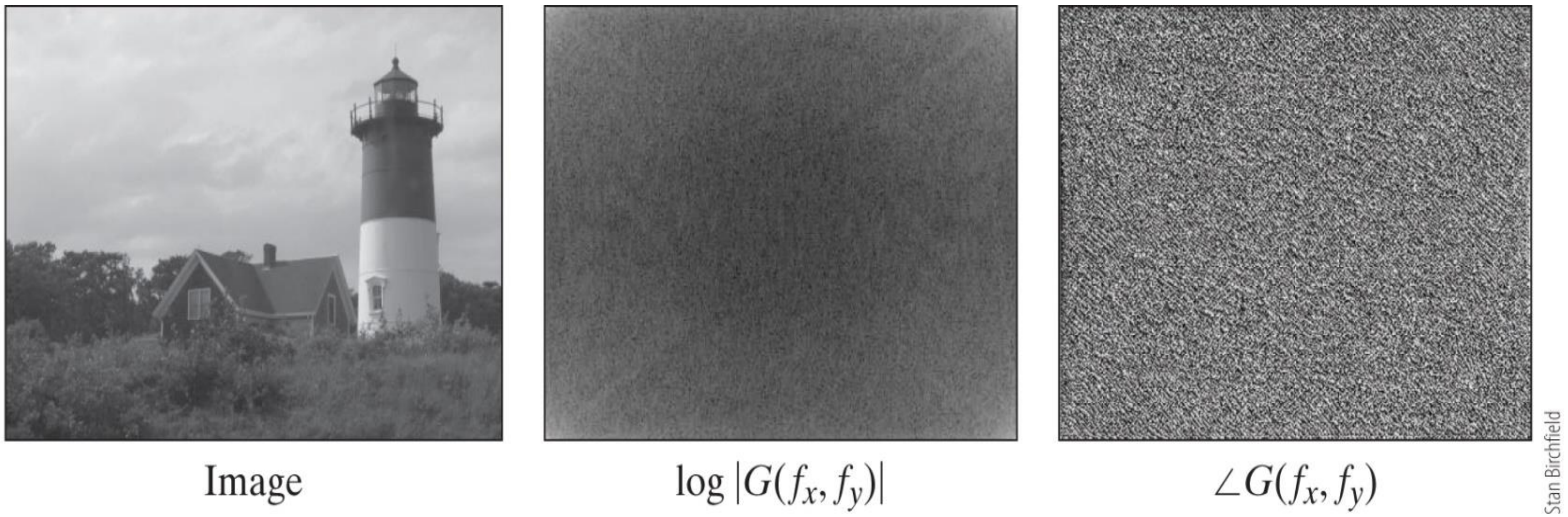
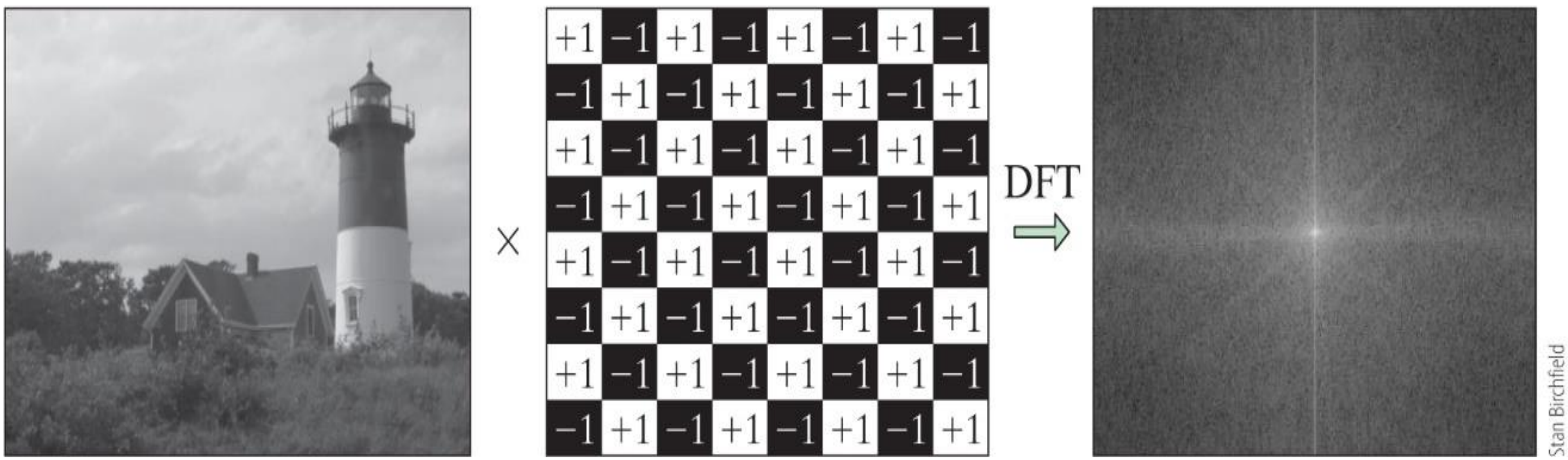


Figure 6.10 An image and its 2D DFT shown as magnitude and phase. (To increase the dynamic range of the display, the log of the magnitude is shown.) The DC component, which is the top-left corner of the magnitude, is difficult to see.

Displaying the 2D DFT (cont.)

Figure 6.11 Multiplying the image by $(-1)^{x+y}$ prior to taking the DFT causes the result to be shifted so that the DC component is in the center. On the right is shown the logarithm of the magnitude of the DFT of the post-multiplied image.



Linear Image Transforms

- Suppose we have a continuous signal $g(\mathrm{x})$ with Fourier transform $G(\mathrm{f})$, and we want to find the Fourier transform of the related signal $g(\mathrm{x}')$, where x’ and x are related by a linear transform:

$$\mathbf{x}' = \mathbf{A}\mathbf{x}$$

- The Fourier transform of the transformed signal is given by:

$$g(\mathbf{x}) \quad \stackrel{\mathcal{F}}{\Longleftrightarrow} \quad G(\mathbf{f})$$

$$g(\mathbf{x}') \quad \stackrel{\mathcal{F}}{\Longleftrightarrow} \quad \frac{1}{\det(\mathbf{A})} G(\mathbf{f}')$$

Frequency Domain Filtering

Frequency-Domain Filtering

- Filtering is used primarily for two applications: restoration and enhancement.

- **Restoration**: the goal is to remove the effects of noise that has degraded the image quality from its original condition.

- **Enhancement:** involves accentuating or sharpening features to make the image more useful, going beyond simply a pure, noise-free image.

Lowpass Filtering

- A **lowpass filter** allows low frequencies to pass through while attenuating high frequencies.

- The **ideal lowpass filter**, also known as the box filter, perfectly passes all frequencies below a certain cutoff, while perfectly attenuating all frequencies above the cutoff.

$$|H_{ilp}(f)| = \begin{cases} 1 & \text{if } f \leq f_c \\ 0 & \text{otherwise} \end{cases}$$

Lowpass Filtering (cont'd)

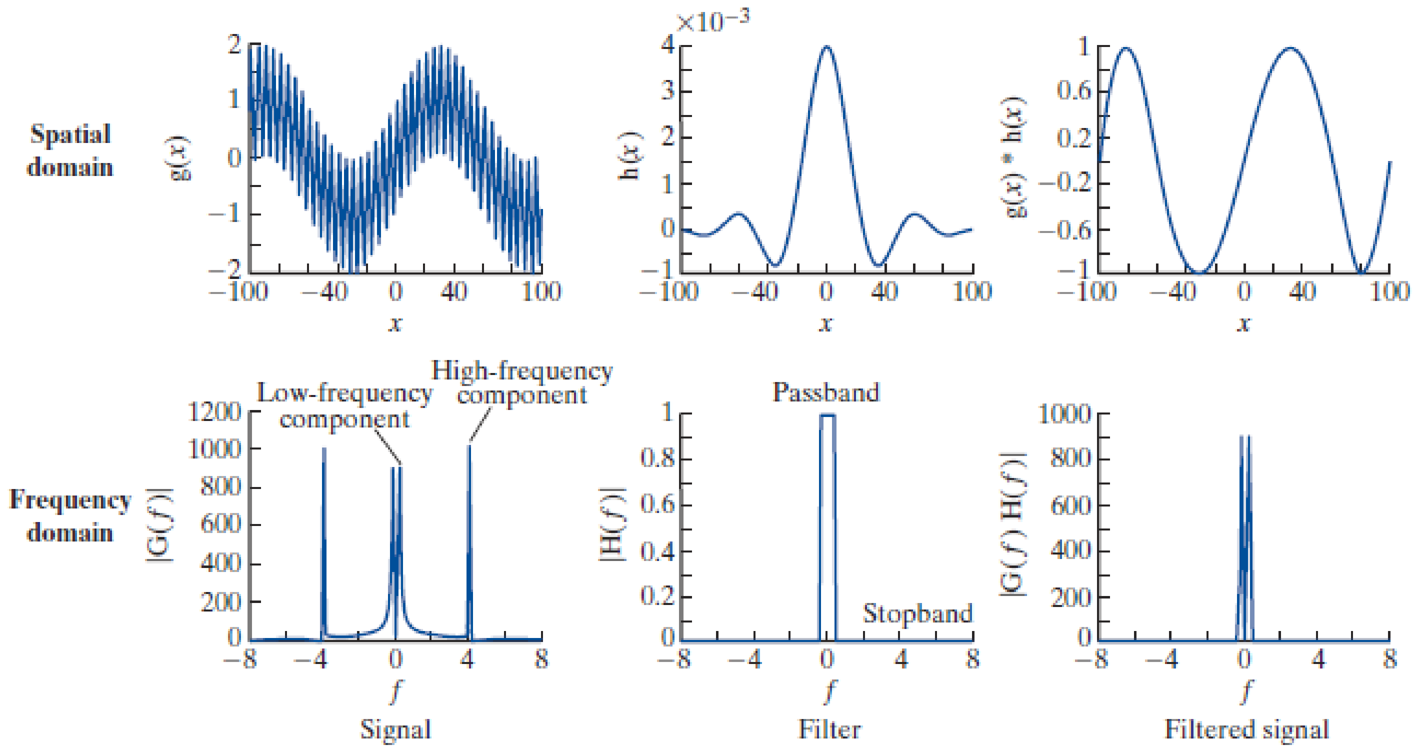
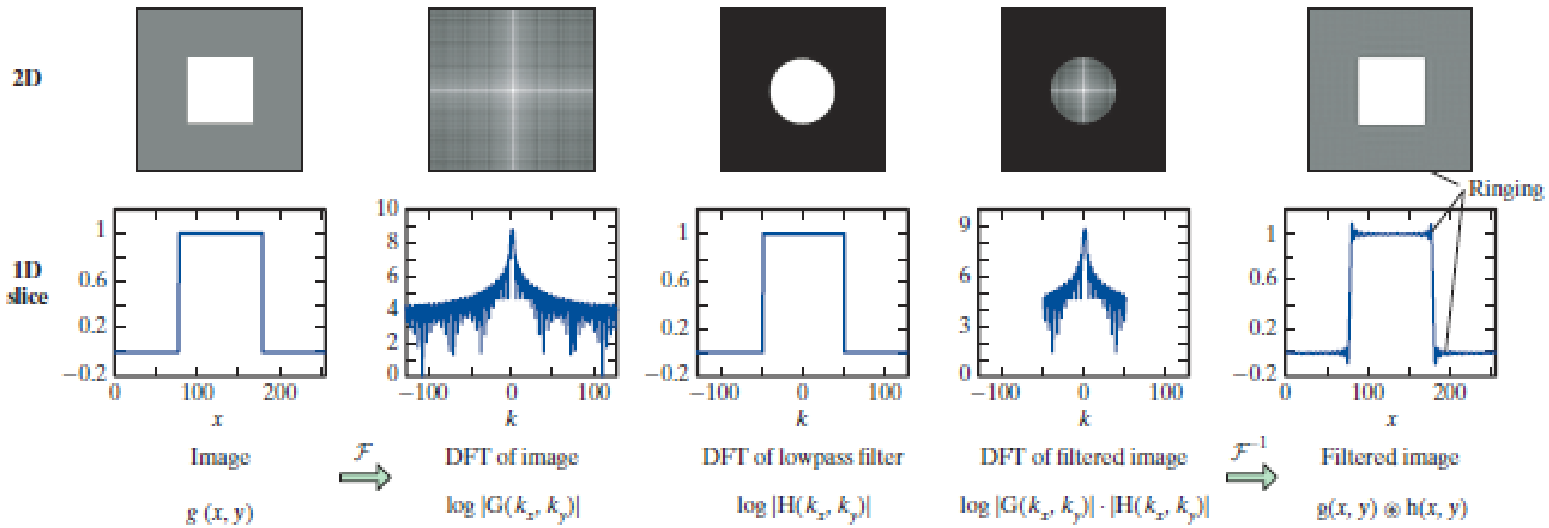


Figure 6.13 The function $\sin(f_1 x) + \sin(f_2 x)$ filtered by an ideal lowpass filter. In the frequency domain, the Fourier transform of the signal is multiplied by a box function. Equivalently, in the spatial domain, the signal is convolved with a sinc function. In this example the filter successfully removes the high-frequency component from the signal, leaving only the low-frequency component.

Lowpass Filtering (cont'd)

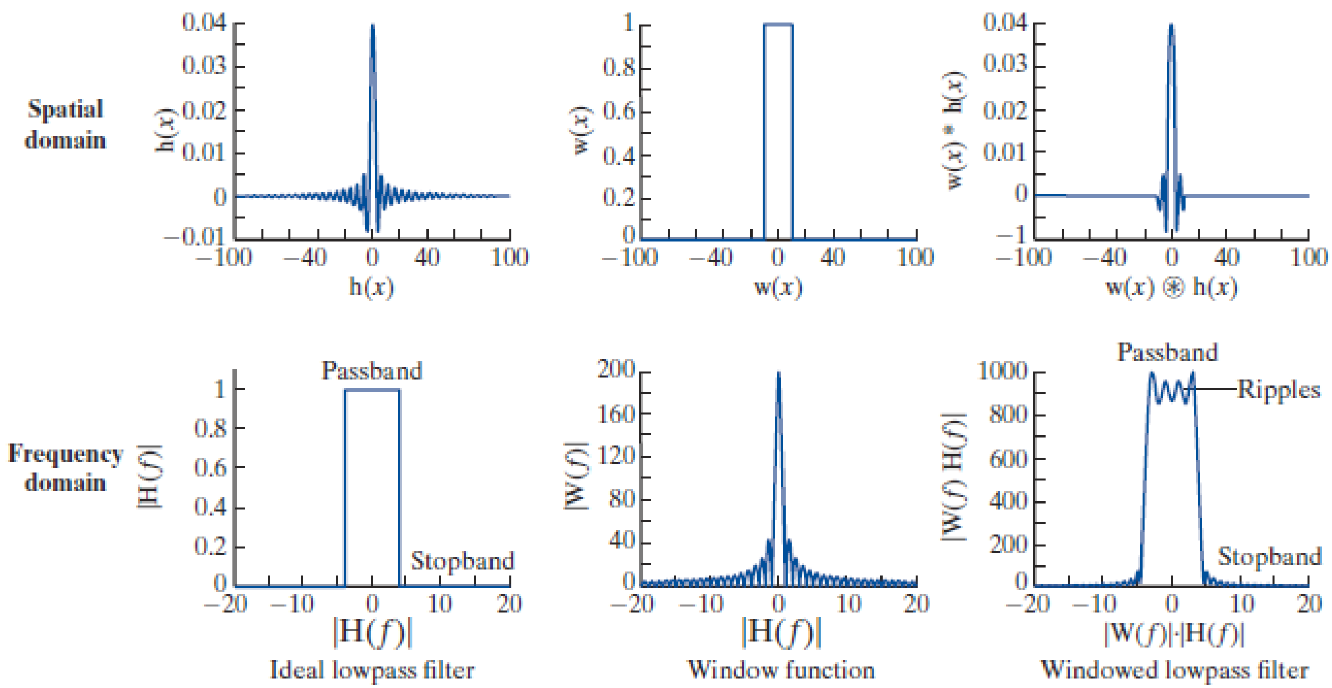
Figure 6.14 The process of frequency-domain filtering. From left to right: The DFT of the image is computed and multiplied by the frequency-domain filter, followed by the inverse DFT to yield the filtered image. Notice in this example that the ideal lowpass filter causes significant ringing in the output.



Windowing

- **Window function**: is a nonnegative function that decreases monotonically from the center, such as the rect function or any of various bell-shaped curves.

Figure 6.15 When the ideal lowpass filter (left) is multiplied by a window function (middle), the resulting filter exhibits ripples (right). Note that the bottom middle plot shows the absolute value of the sinc function.



Gaussian Lowpass Filter

- **Gaussian lowpass filter:**

$$|H_{glp}(f)| = e^{-f/2{f_c}^2}$$

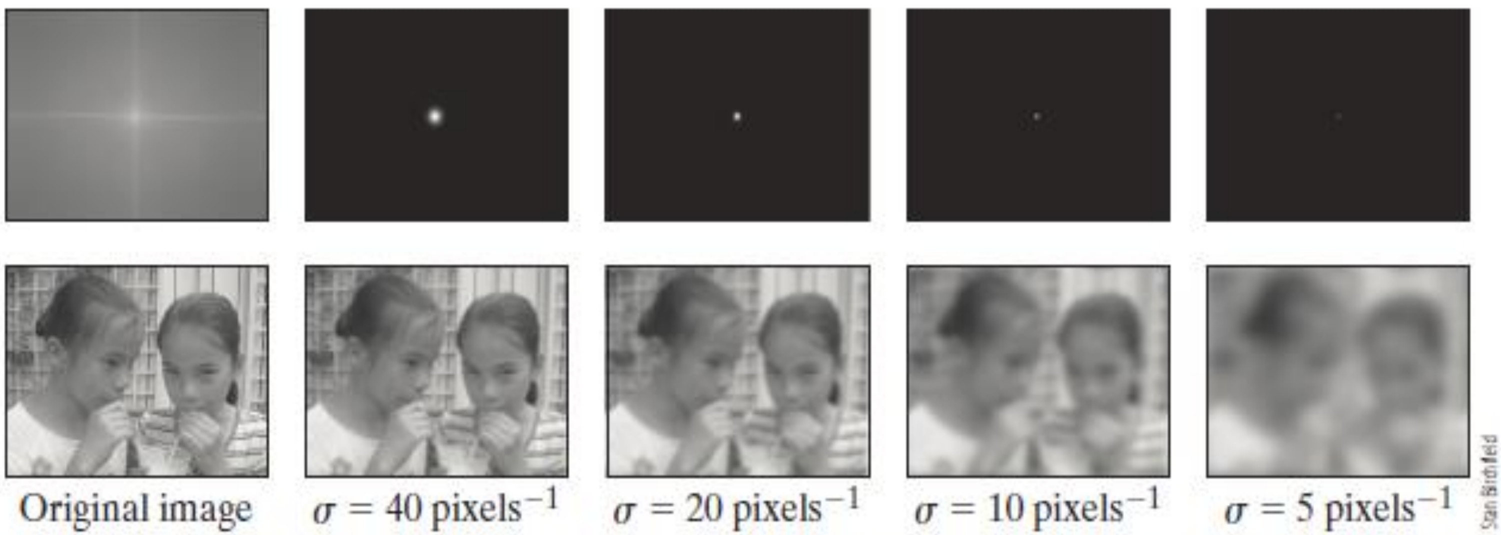


Figure 6.16 An image, and the result of Gaussian low-pass filtering in the frequency domain with different variances. The top row shows the DFT of the image and the magnitude of the frequency response of each filter. The smoothed images are the inverse DFT of the multiplication of the image DFT with the various filter frequency responses. Note that a large variance in the frequency domain yields less smoothing, whereas a small variance yields more smoothing.

Butterworth Lowpass Filter

- **Elliptic filter**: a general type of filter that allows the designer to independently specify the amount of ripple in the passband and stopband.
- **Chebyshev filter:** as the amount of ripple goes to zero in the passband or stopband, the elliptic filter is well approximated by a **Chebyshev filter**.
- If the ripple goes to zero in both the passband and stopband, the elliptic filter approximates a **Butterworth filter**.

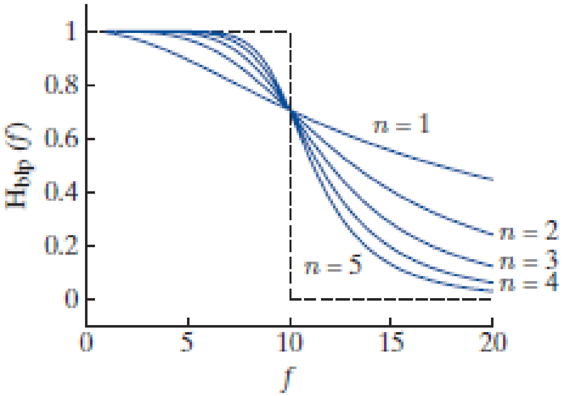
Butterworth Lowpass Filter (cont'd)

- The Butterworth filter is also known as the *maximally flat* filter.

- The magnitude-squared of the Butterworth lowpass filter of order *n* is given by:

$$|H_{blp}(f)|^2 = \frac{1}{1 + (f/f_c)^{2n}}$$

Figure 6.17 The magnitude of the Butterworth lowpass filter for $n = 1$ to $n = 5$ (solid lines). As n increases, the Butterworth response approaches the ideal lowpass filter (dashed line).



Lanczos Filter

When the sinc function is multiplied by the first lobe of another sinc function, the result is the **Lanczos filter**.

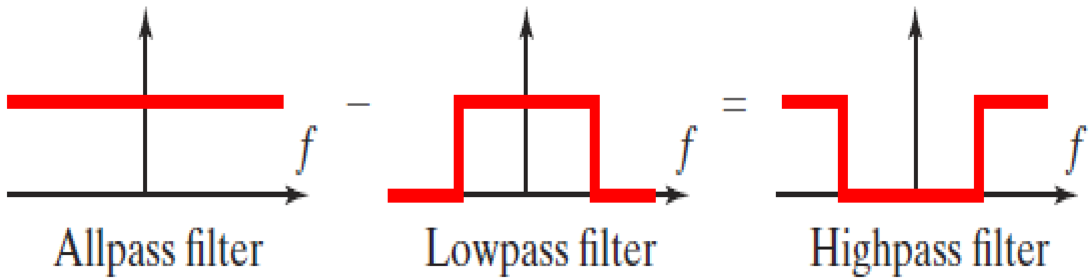
- High-quality filter that is widely used in image processing.
- Particularly used for smoothing an image before downsampling.

Highpass Filtering

- In the frequency domain, the magnitude of a **highpass filter** is the magnitude of an allpass filter minus the magnitude of the corresponding lowpass filter.

$$|H_{highpass}(f)| = 1 - |H_{lowpass}(f)|$$

Figure 6.18 A highpass filter is the allpass filter minus a lowpass filter



Highpass Filtering (cont'd)

Figure 6.19 An image, and the result of Gaussian high-pass filtering in the frequency domain with different variances. The top row shows the DFT of the image and the magnitude of the frequency response of each filter. Bright values indicate frequencies that are passed, whereas dark values indicate frequencies that are attenuated.

Bandpass Filtering

- A **bandpass filter** rejects both low and high frequencies, instead passing only frequencies in a certain band.

- The ideal bandpass filter:

$$|H(f)| = \begin{cases} 1 & \text{if } f_{lo} \le f \le f_{hi} \\ 0 & \text{otherwise} \end{cases}$$

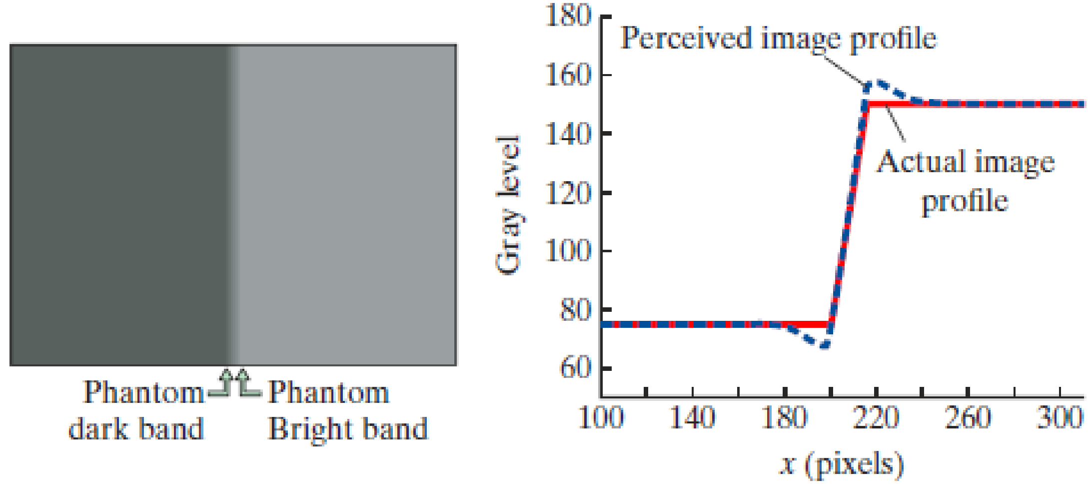
- The most common bandpass filter for image processing is the **Laplacian of Gaussian (LoG) filter.**

$$|H(f)| = -f^2 e^{-f/2f_c^2}$$

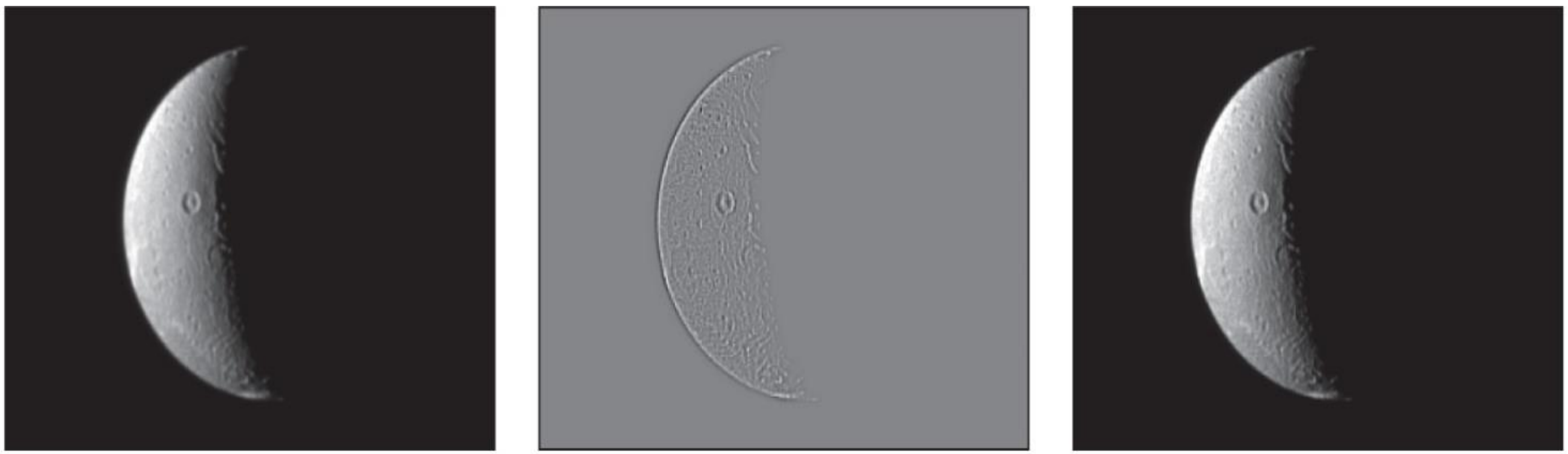
Unsharp Masking and Highboost Filtering

This approach takes advantage of a peculiarity of the human visual system, namely that neurons in the retina distort the intensity values based on neighboring intensities.

Figure 6.20 The Mach bands illusion. Left: Image consisting of dark and light regions, with a linear transition between them. The human visual system hallucinates a dark band left of the transition and a bright band right of the transition. Right: 1D slice through the image, showing the actual graylevel function and the perceived function.



Unsharp Masking and Highboost Filtering (cont.)

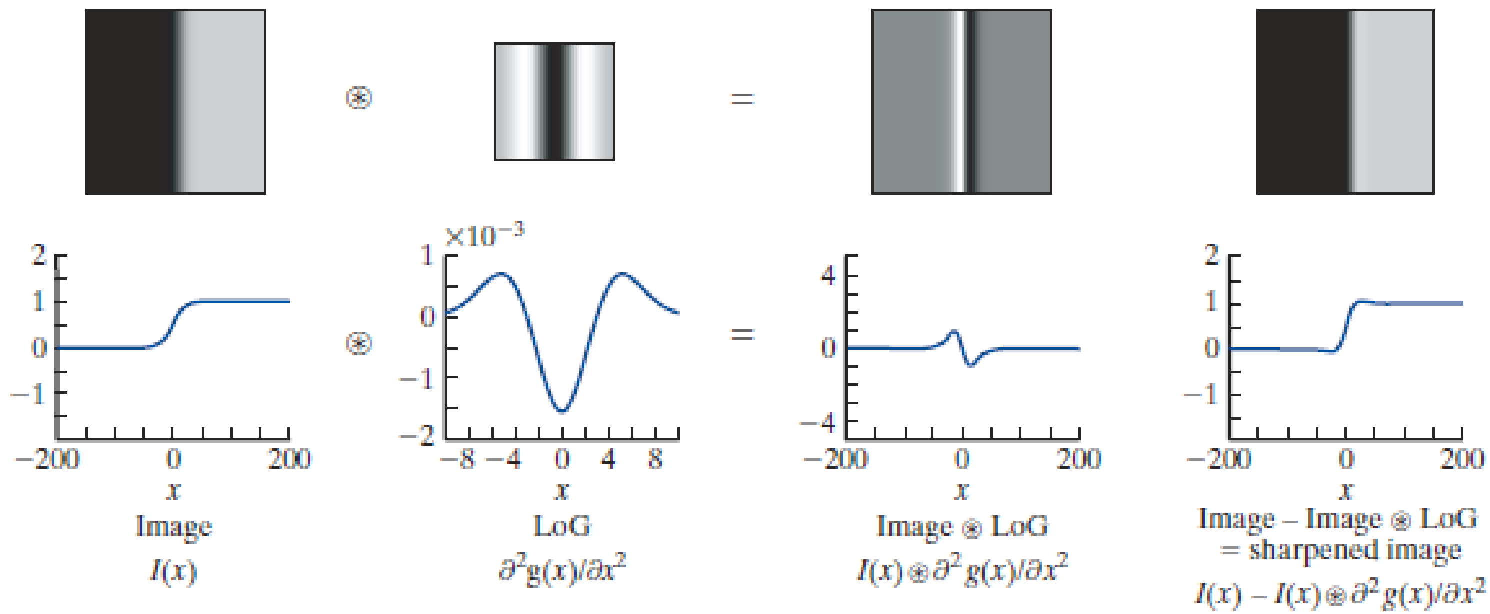


Original image Image minus lowpass Sharpened image NASA

Figure 6.21 From left to right: An image of Saturn's moon Dione, the result of subtracting the low-pass filtered version of the image from itself, and the sharpened image resulting from adding this subtraction back to the original image.

Unsharp Masking and Highboost Filtering (cont.)

Figure 6.22 The process of image sharpening: The image is convolved with the LoG, and the result is subtracted from the original image. The edge in the right column appears sharper than that in the left column.



Questions?