Last time:

Last time:

Lowpass filters:

Highpass filters:

Last time:

Lowpass filters: These are filters that allow only the low frequencies and take out the high frequencies

Highpass filters: These are filters that allow only the high frequencies and take out the low frequencies

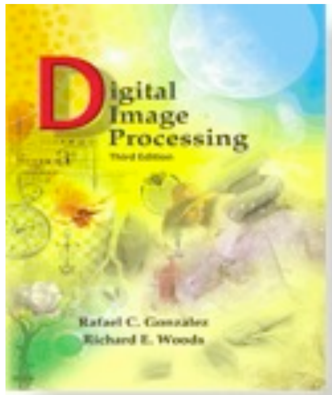
Last time:

Lowpass filters: These are filters that allow only the low frequencies and take out the high frequencies

low frequencies: smoothing

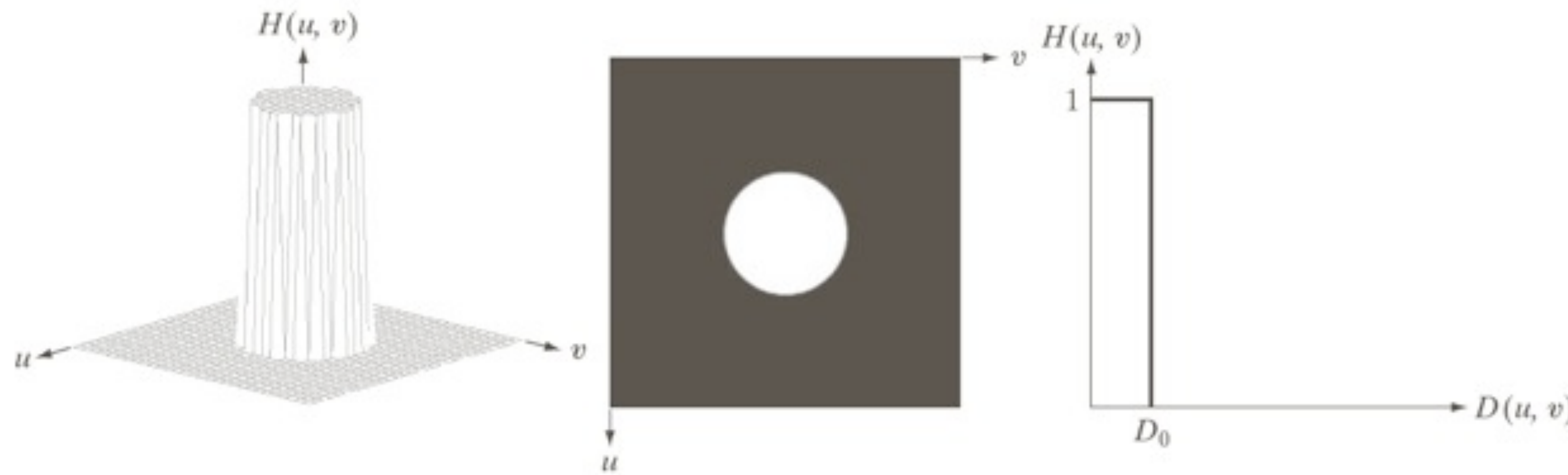
Highpass filters: These are filters that allow only the high frequencies and take out the low frequencies

high frequencies: sharpening



Last time: filtering in the frequency domain - Ideal lowpass filters (ILPF)

1. Transform image to frequency space (matlab fft2)
2. Centre the spectrum (fftshift)
3. Create filter H in the frequency domain
4. Apply the filter H*F (pointwise)
5. Reconstruct the image (ifft2)



a b c

FIGURE 4.40 (a) Perspective plot of an ideal lowpass-filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

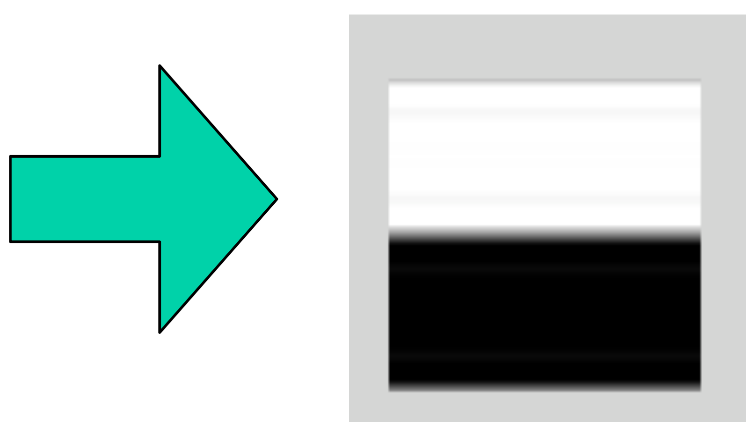
$$H(u,v) = 1, \quad \text{if } D(u,v) \leq D_0$$
$$H(u,v) = 0 \quad \text{if } D(u,v) > D_0$$

$$D(u,v) = \sqrt{(u - P/2)^2 + (v - Q/2)^2}$$

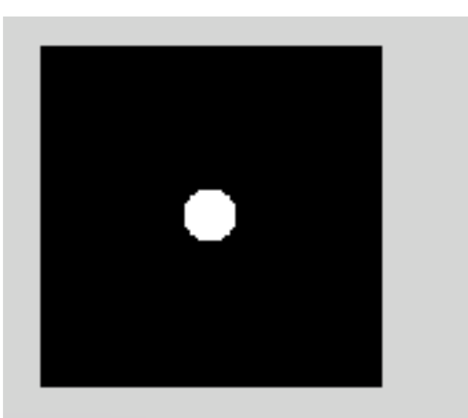
distance from the center of the image

The actual result might depend on the periodicity pattern of the image.

Image

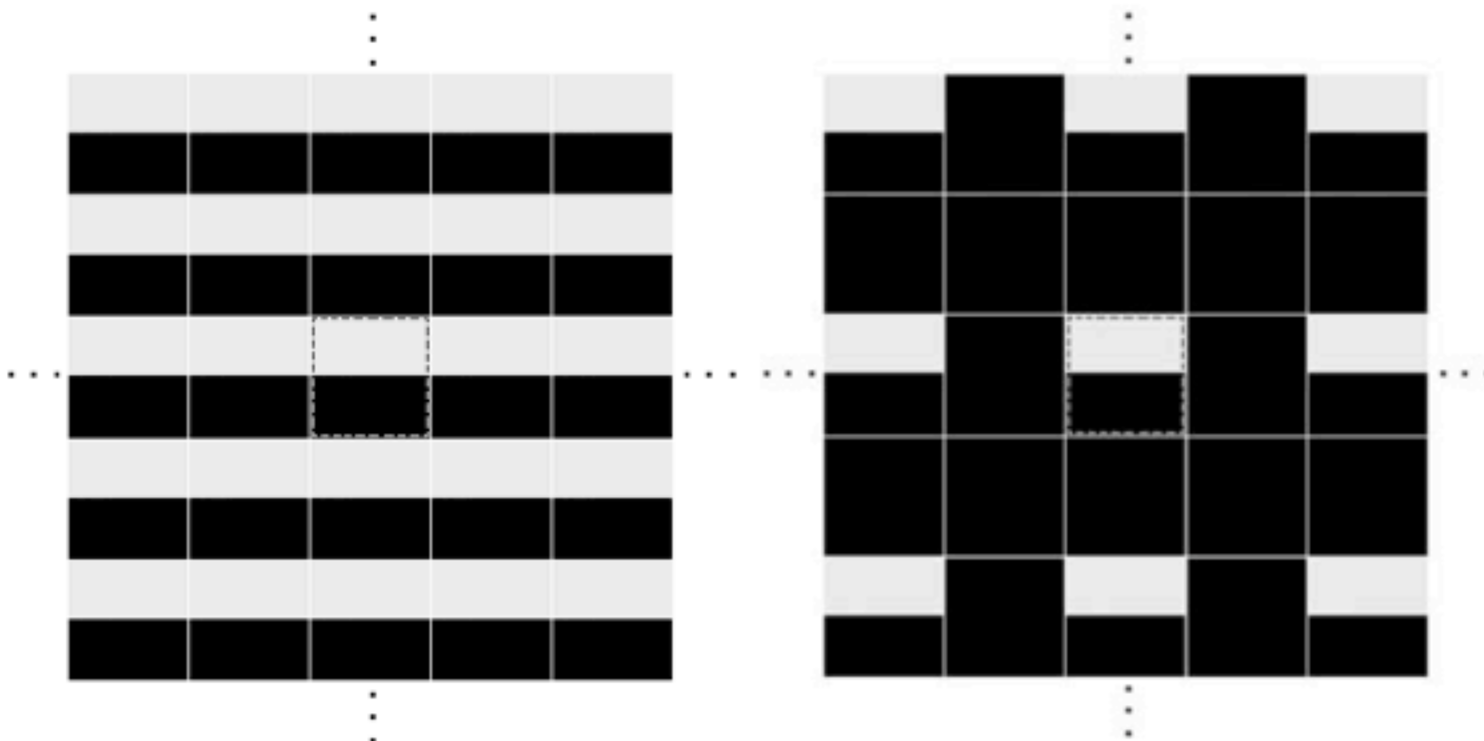


Result: the blurring occurs only on the vertical dimension, and is not uniform on both spatial directions.



ILPF

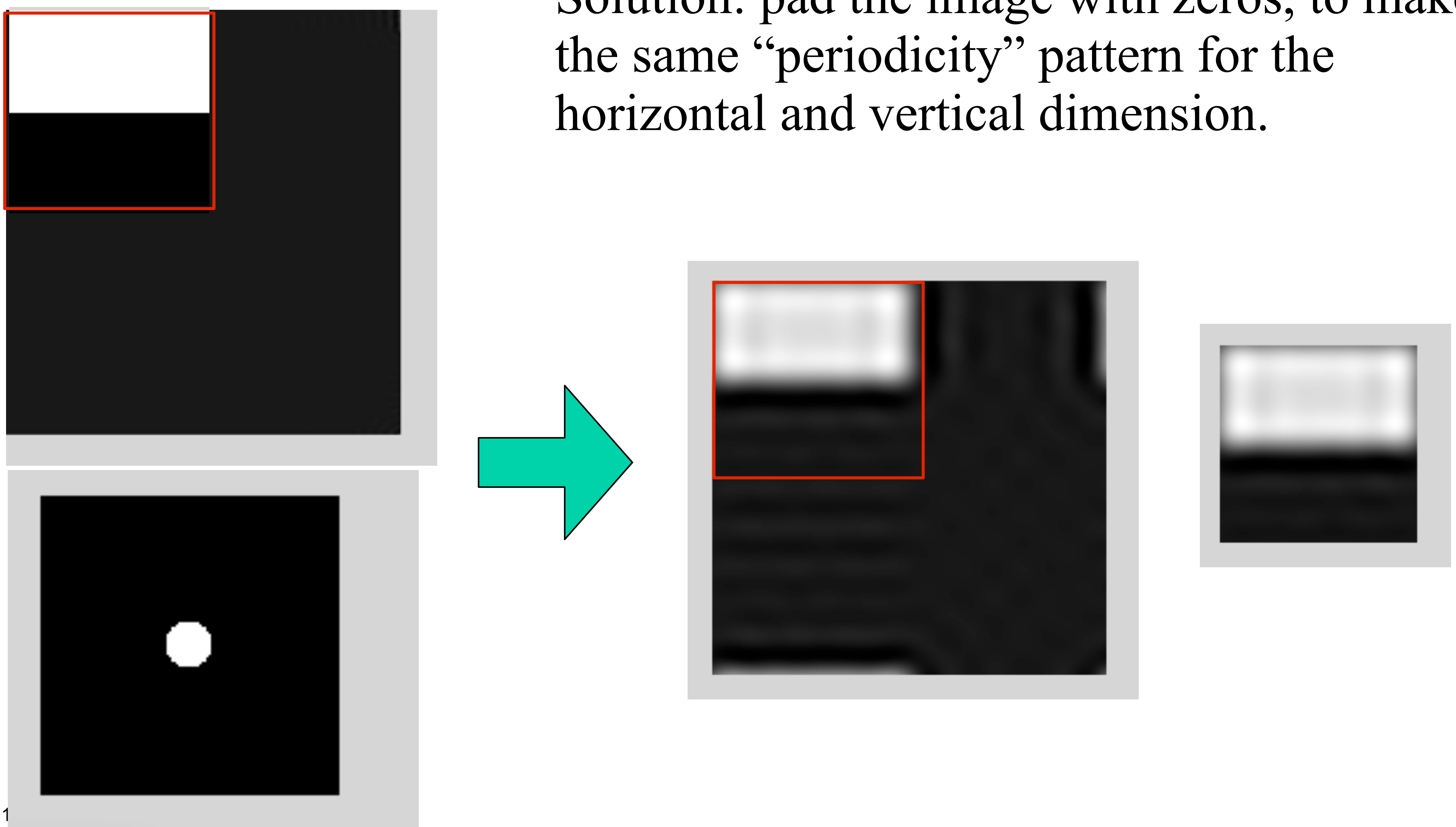
A similar result would be obtained if the image was rotated 90 degrees.

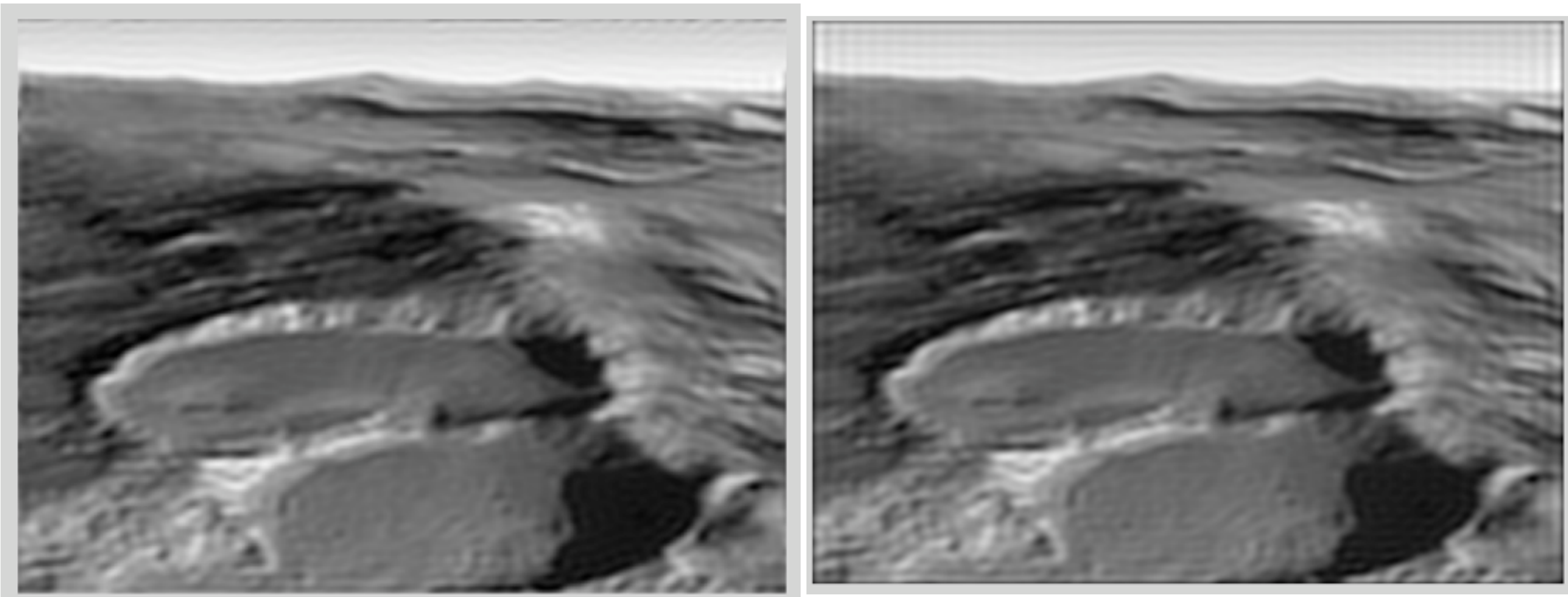


a b

FIGURE 4.33 2-D image periodicity inherent in using the DFT. (a) Periodicity without image padding. (b) Periodicity after padding with 0s (black). The dashed areas in the center correspond to the image in Fig. 4.32(a). (The thin white lines in both images are superimposed for clarity; they are not part of the data.)

Solution: pad the image with zeros, to make the same “periodicity” pattern for the horizontal and vertical dimension.





Last time’s example without padding (left) and with padding (right). Note the behaviour near the border of the image.

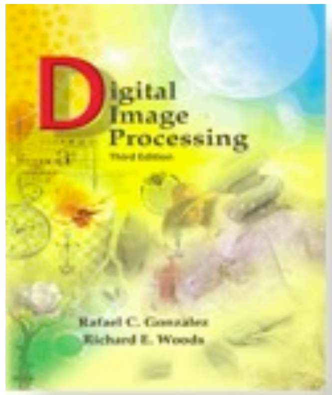
Main steps of filtering in the frequency domain:

Assume *f(x,y)* (image), *h(x,y)* (filter), in the spatial domain given:

1. Pad *f(x,y)* to obtain *fp(x,y)*, and similarly for h, hp
2. Compute dft of fp, hp (matlab: fft2)
3. Center the transforms (matlab: fftshift), Fp, Hp
4. Multiply *G(u,v) = Fp(u,v) Hp(u,v)*

5. Obtain the processed image *gp(x,y) = Real(ifft2(fftshift(G(u,v)))* (extract the desired subportion).

NB. We take the real part to remove spurious imaginary parts that come from our computations (rounding etc.)



How to choose the cutoff frequency D_0 for ILPF?

We look at the image power:

1. Construct the Power spectrum of the padded image, after centering the spectrum

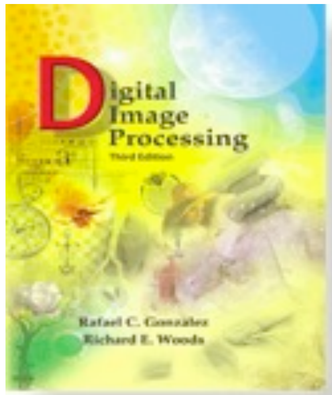
2. Compute $$P_T = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} P(u,v)$$

3. Choose a percent of the image power, 0<alpha<=100:

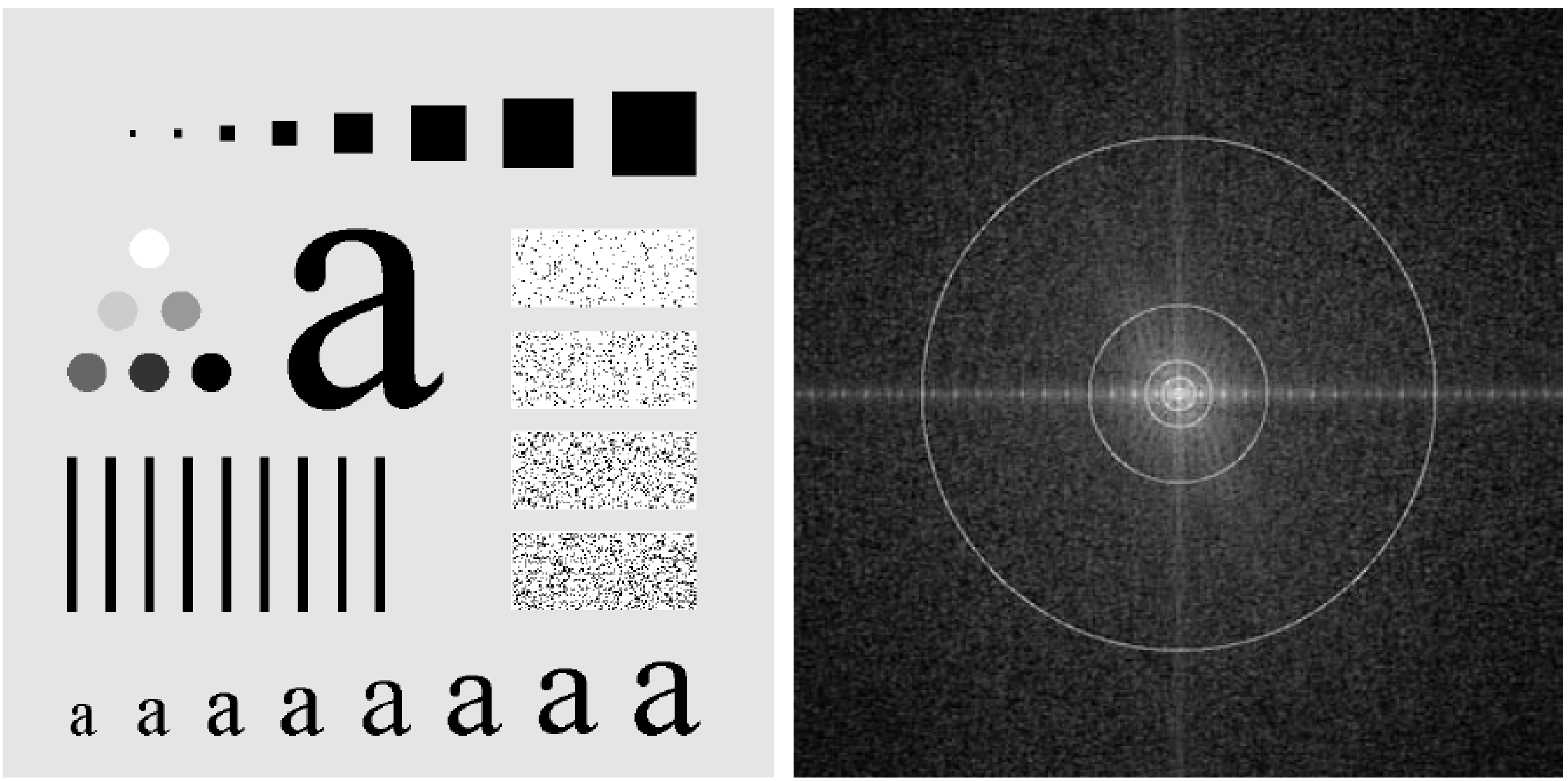
If the DFT has been centered, a disc with radius D_0 includes alpha percent of the power if

$$\alpha = \frac{100}{P_T} \sum_{u} \sum_{v} P(u, v)$$

where the sum is carried over all the indices (u,v) so that the sum is less or equal to alpha.

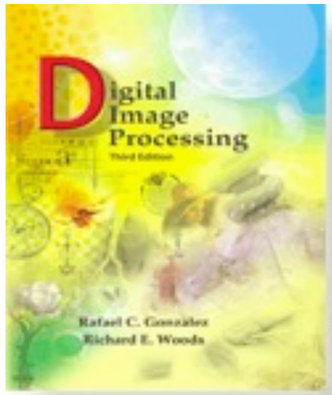


matlab example:



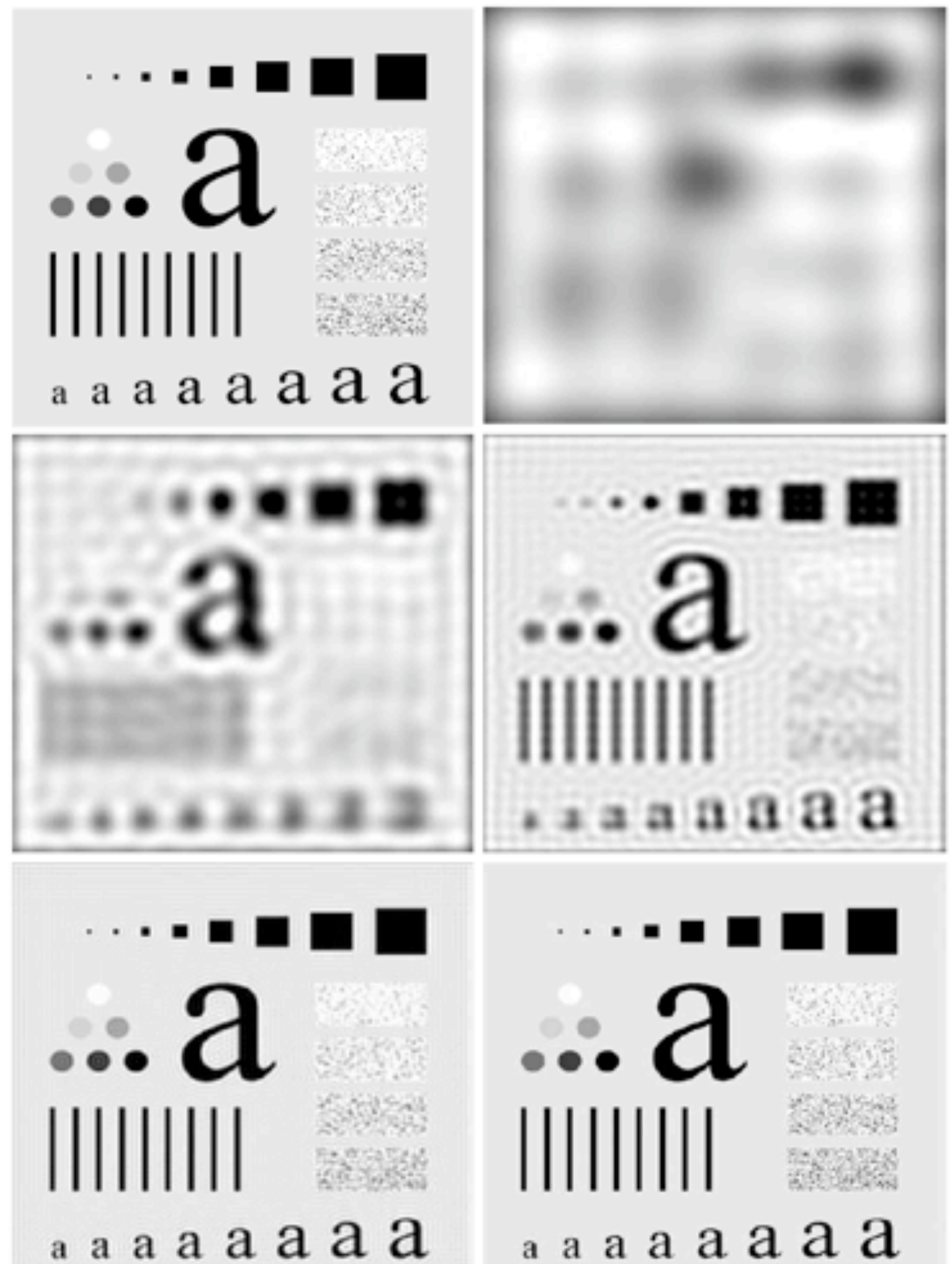
a b

FIGURE 4.41 (a) Test pattern of size 688 × 688 pixels, and (b) its Fourier spectrum. The spectrum is double the image size due to padding but is shown in half size so that it fits in the page. The superimposed circles have radii equal to 10, 30, 60, 160, and 460 with respect to the full-size spectrum image. These radii enclose 87.0, 93.1, 95.7, 97.8, and 99.2% of the padded image power, respectively.



Chapter 4

Filtering in the Frequency Domain



a	b
c	d
e	f

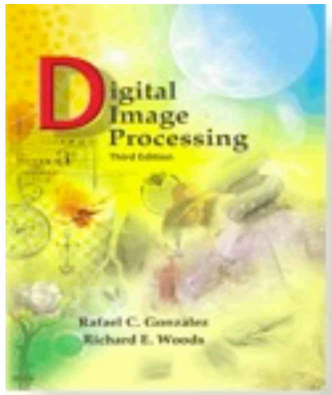
FIGURE 4.42 (a) Original image. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.41(b). The power removed by these filters was 13, 6.9, 4.3, 2.2, and 0.8% of the total, respectively.

Lowpass filters:

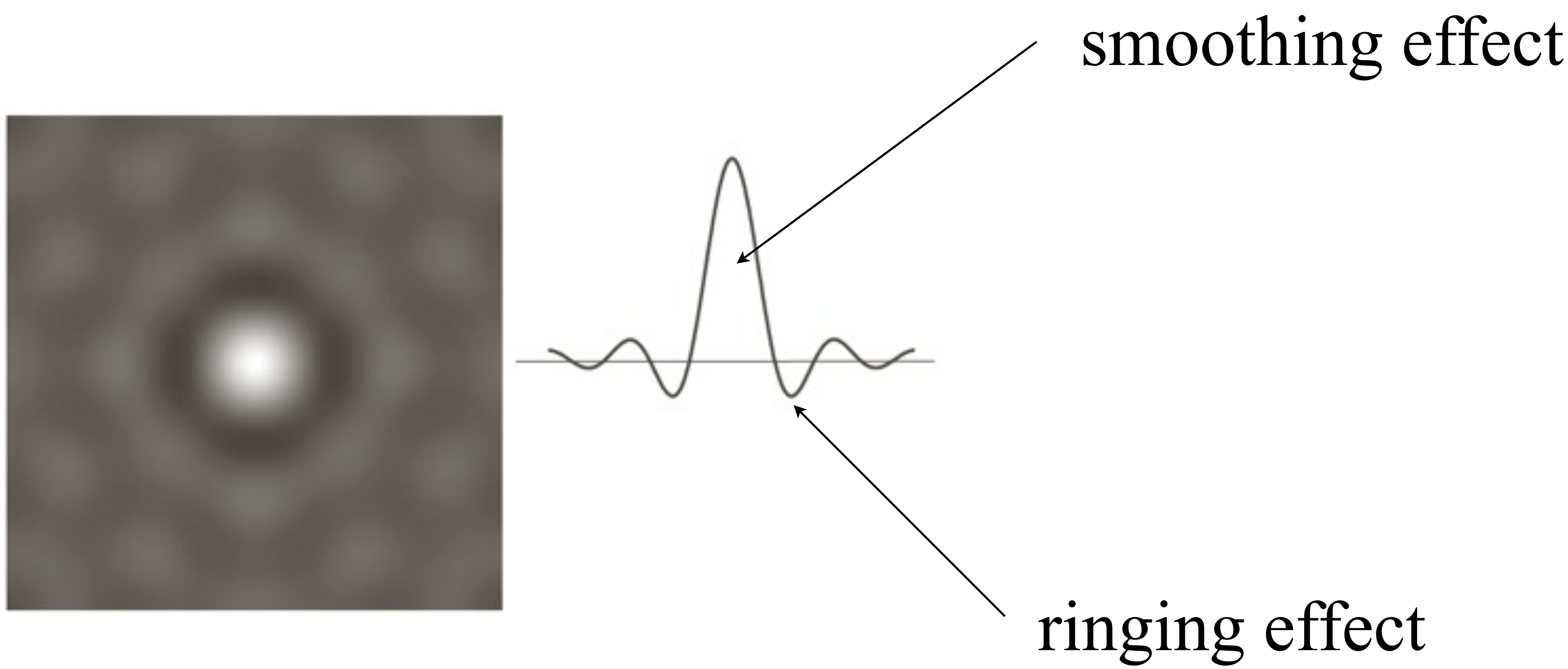
- Ideal lowpass filter

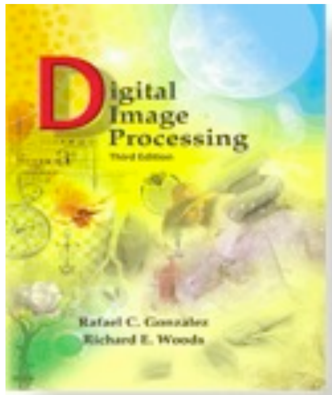
- Gaussian lowpass filter

- Butterworth lowpass filter



We have seen that the ILPF causes ringing effect in the image (because its inverse transform is a sinc function).



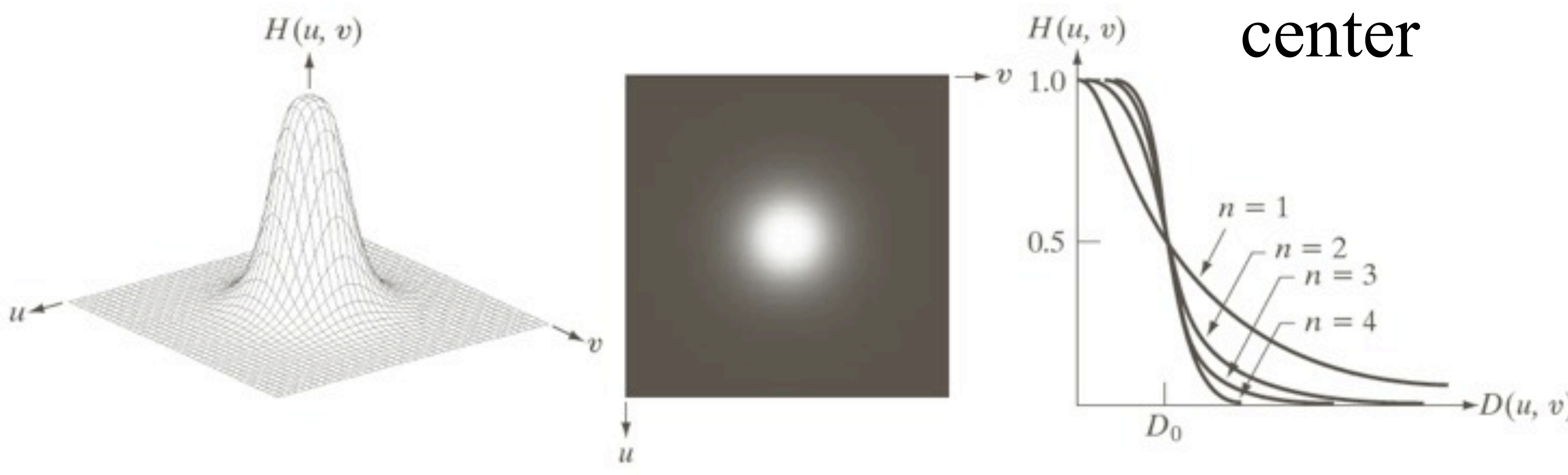


The **Butterworth** filter has the overall effect of a step function, but it has smudged edges, with the effect of reducing the ringing effect

cutoff frequency

$$H(u,v) = \frac{1}{1 + (D(u,v)/D_0)^{2n}}$$

distance from the center



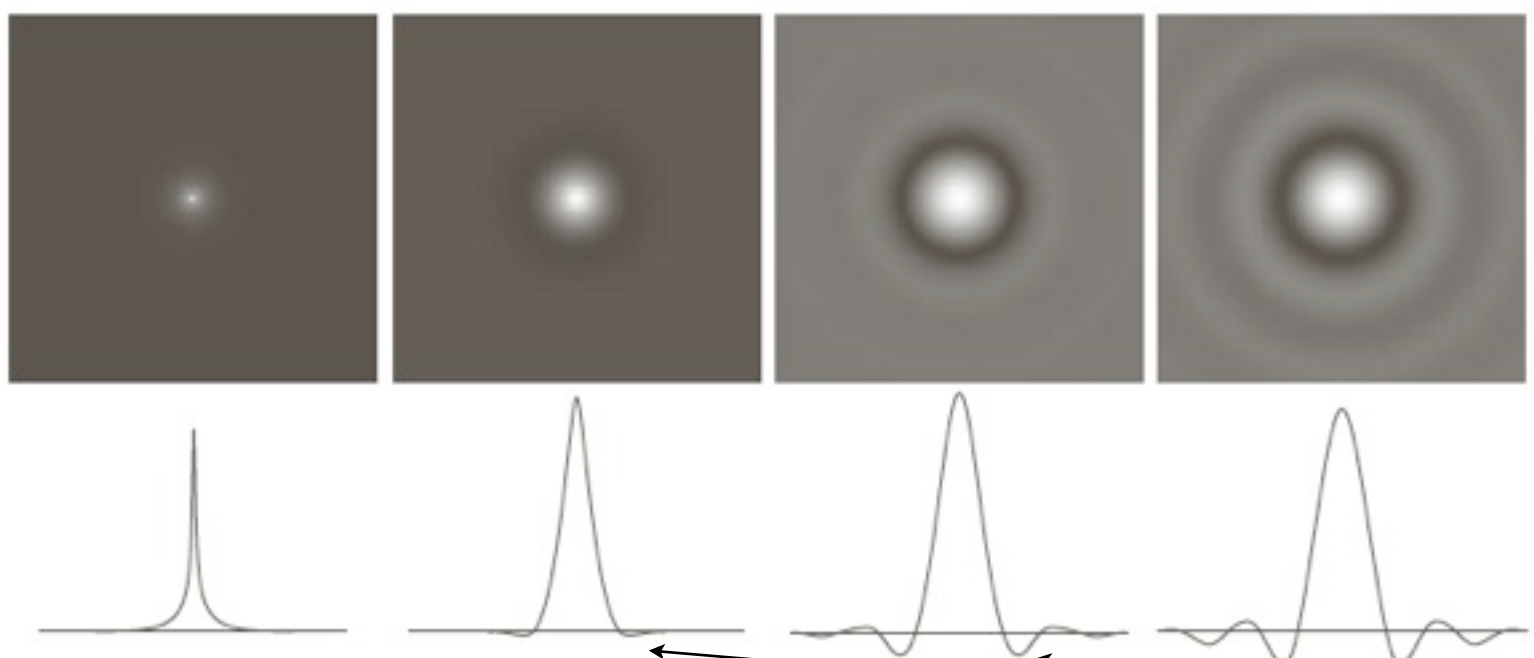
a b c

FIGURE 4.44 (a) Perspective plot of a Butterworth lowpass-filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

n=*1*: no ringing in the spatial domain

n=*2*: almost imperceptible ringing

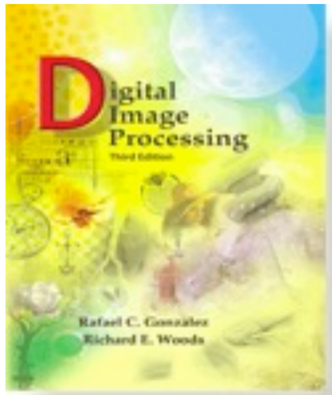
ringing can appear for higher values of *n*



a b c d

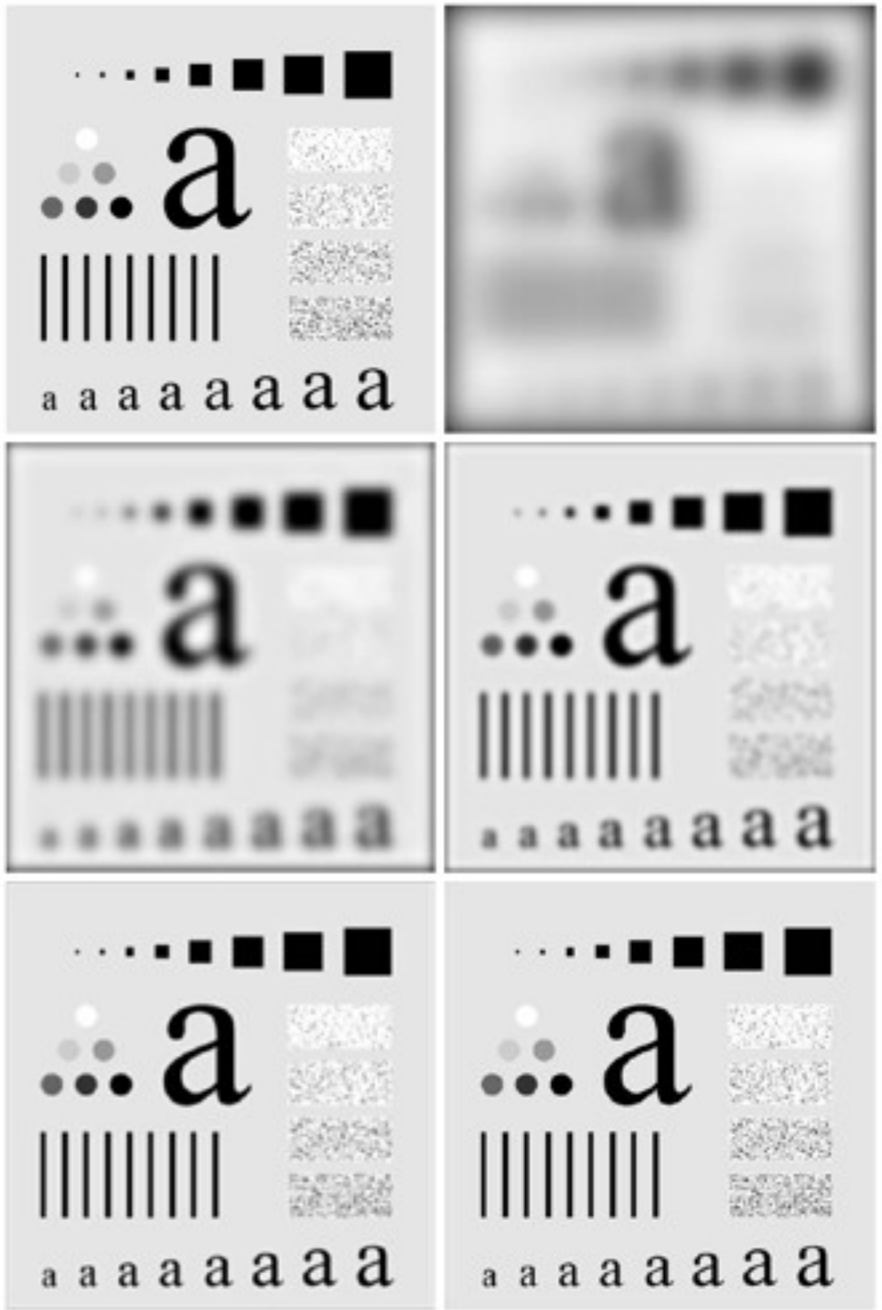
FIGURE 4.46 (a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding intensity profiles through the center of the filters (the size in all cases is 1000 × 1000 and the cutoff frequency is 5). Observe how ringing increases as a function of filter order.

The negative values cause the ringing



Chapter 4

Filtering in the Frequency Domain



a b
c d
e f

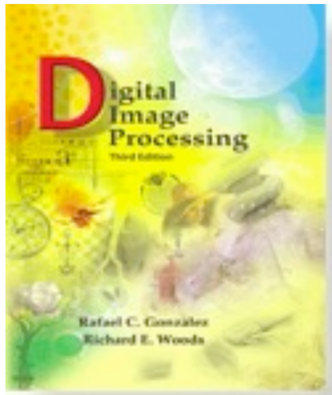
FIGURE 4.45 (a) Original image. (b)–(f) Results of filtering using BLPFs of order 2, with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Fig. 4.42.

Lowpass filters:

- Ideal lowpass filter

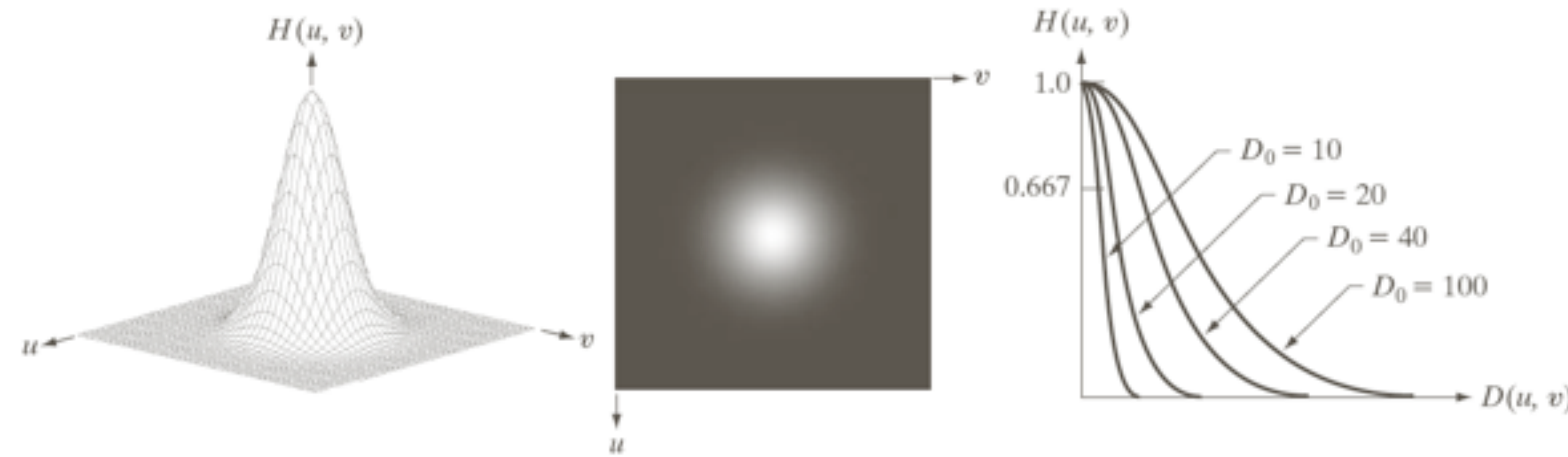
- Gaussian lowpass filter

- Butterworth lowpass filter



Gaussian lowpass filters

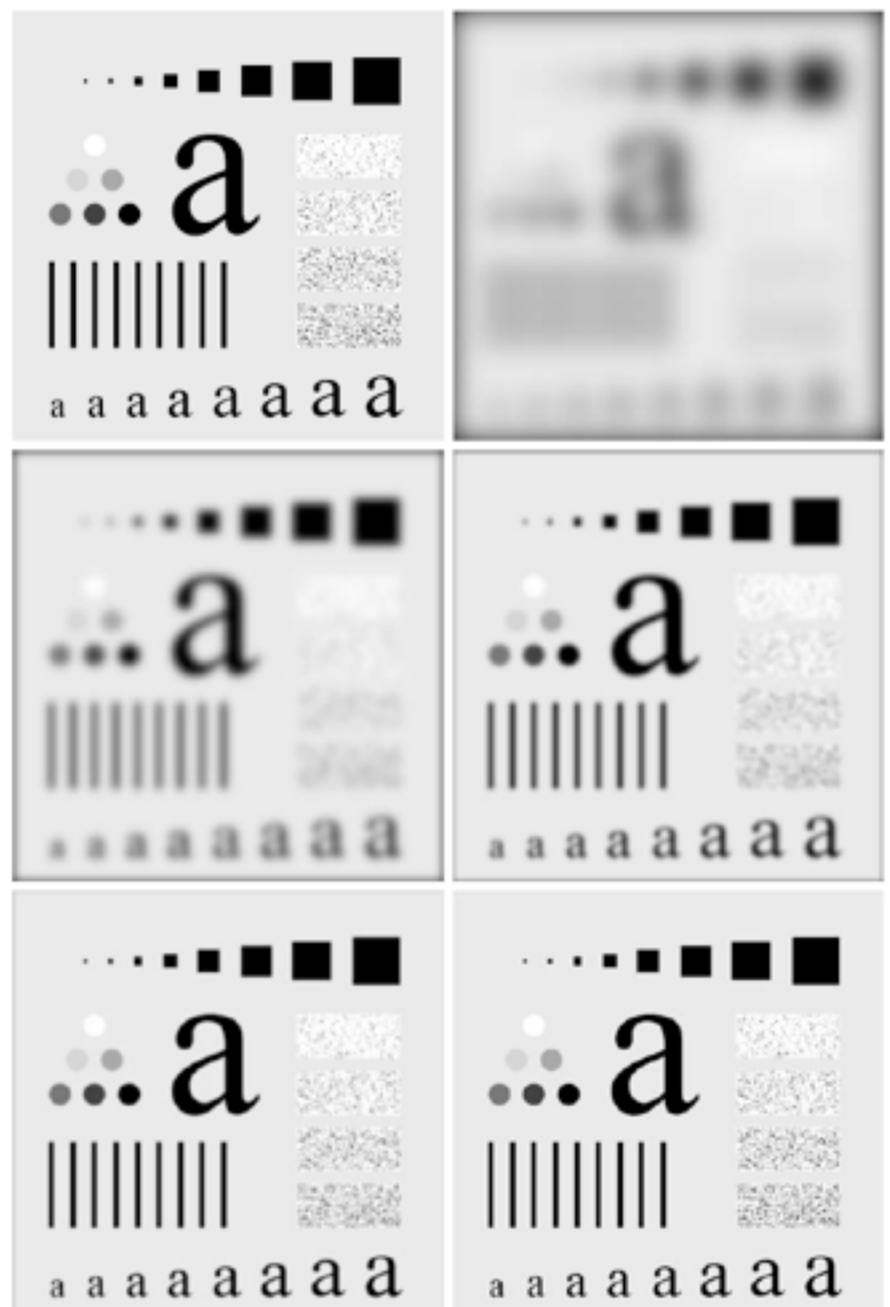
$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$



(NB. Compare the 2D0^2 with 2sigma^2)

Gaussian functions, as the Dirac comb, also has the property that it becomes a new Gaussian under Fourier transform (and the inverse transform).

As a consequence, its spatial transform will have no negative values, hence no ringing effect.



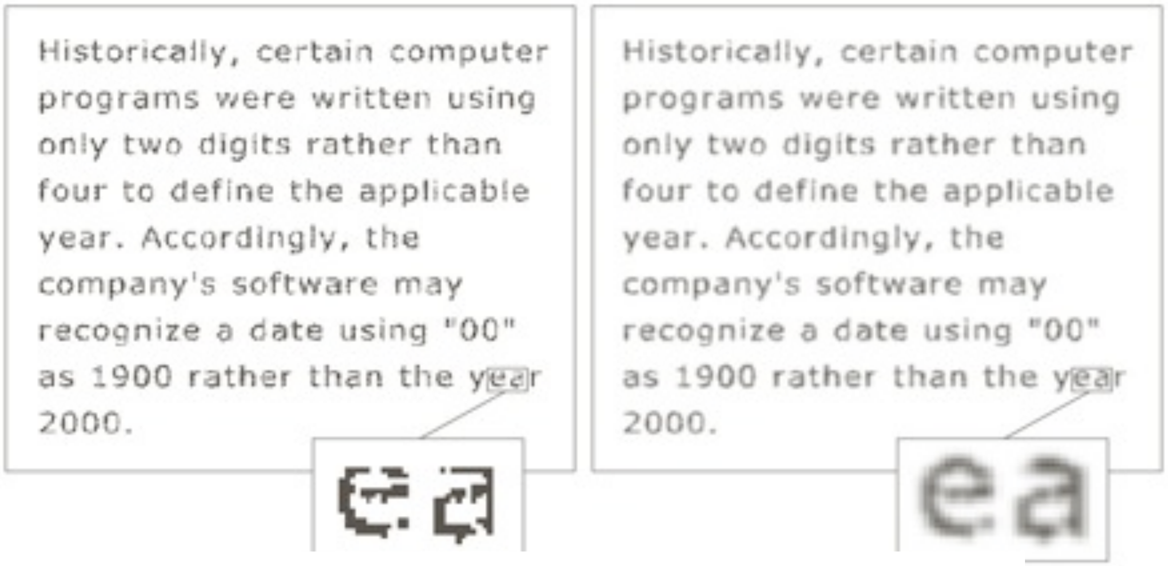
a b
c d
e f

FIGURE 4.48 (a) Original image. (b)–(f) Results of filtering using GLPFs with cutoff frequencies at the radii shown in Fig. 4.41. Compare with Figs. 4.42 and 4.45.

TABLE 4.4
Lowpass filters. D_0 is the cutoff frequency and n is the order of the Butterworth filter.

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

Some applications in characters recognition and printing

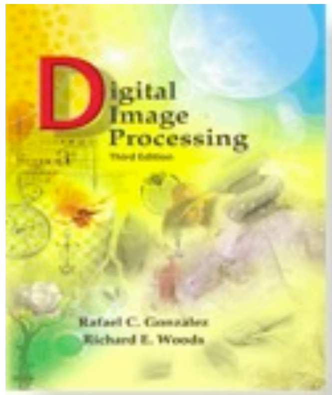


Highpass filters as sharpening filters

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

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

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



Chapter 4

Filtering in the Frequency Domain

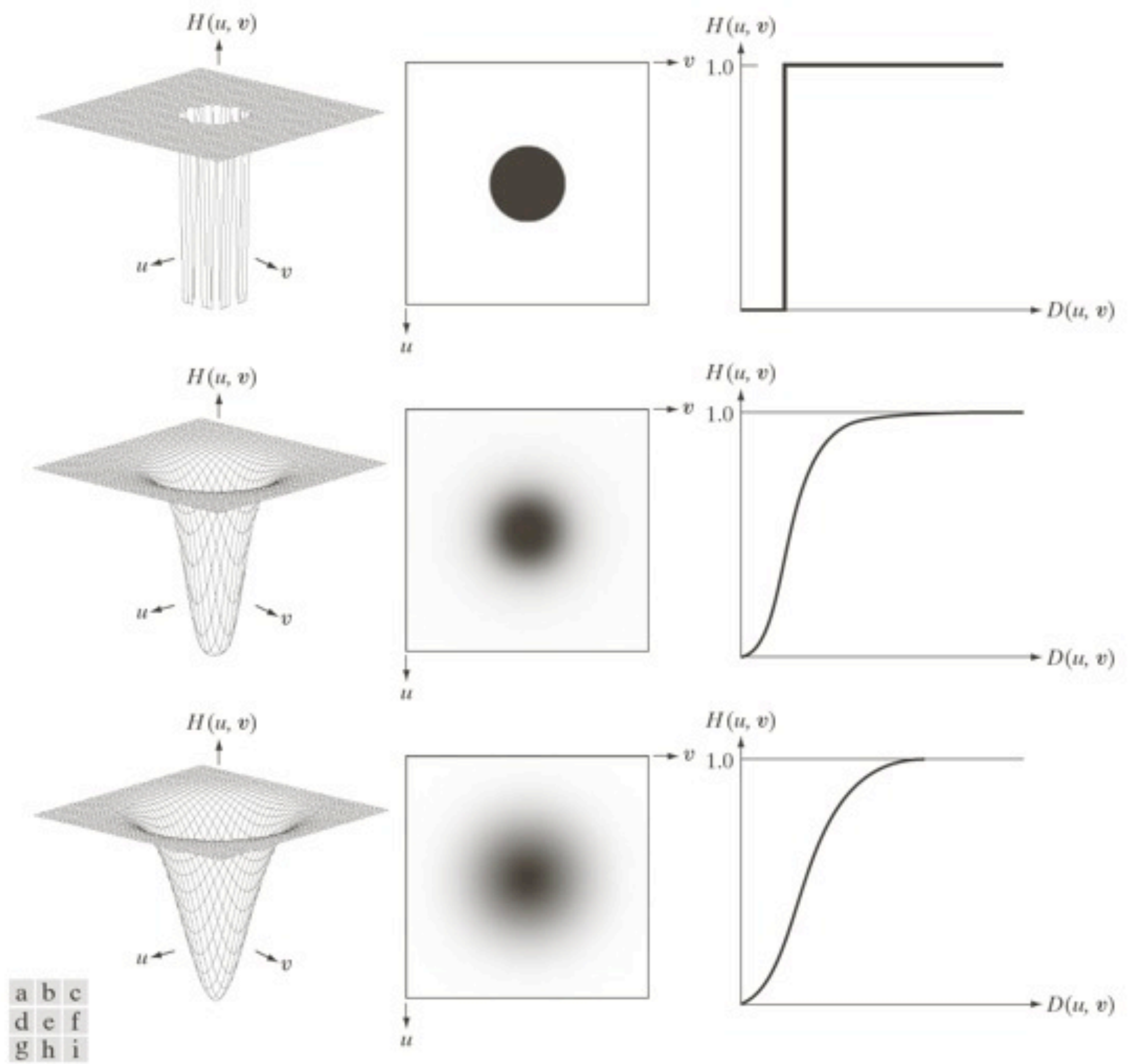


FIGURE 4.52 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

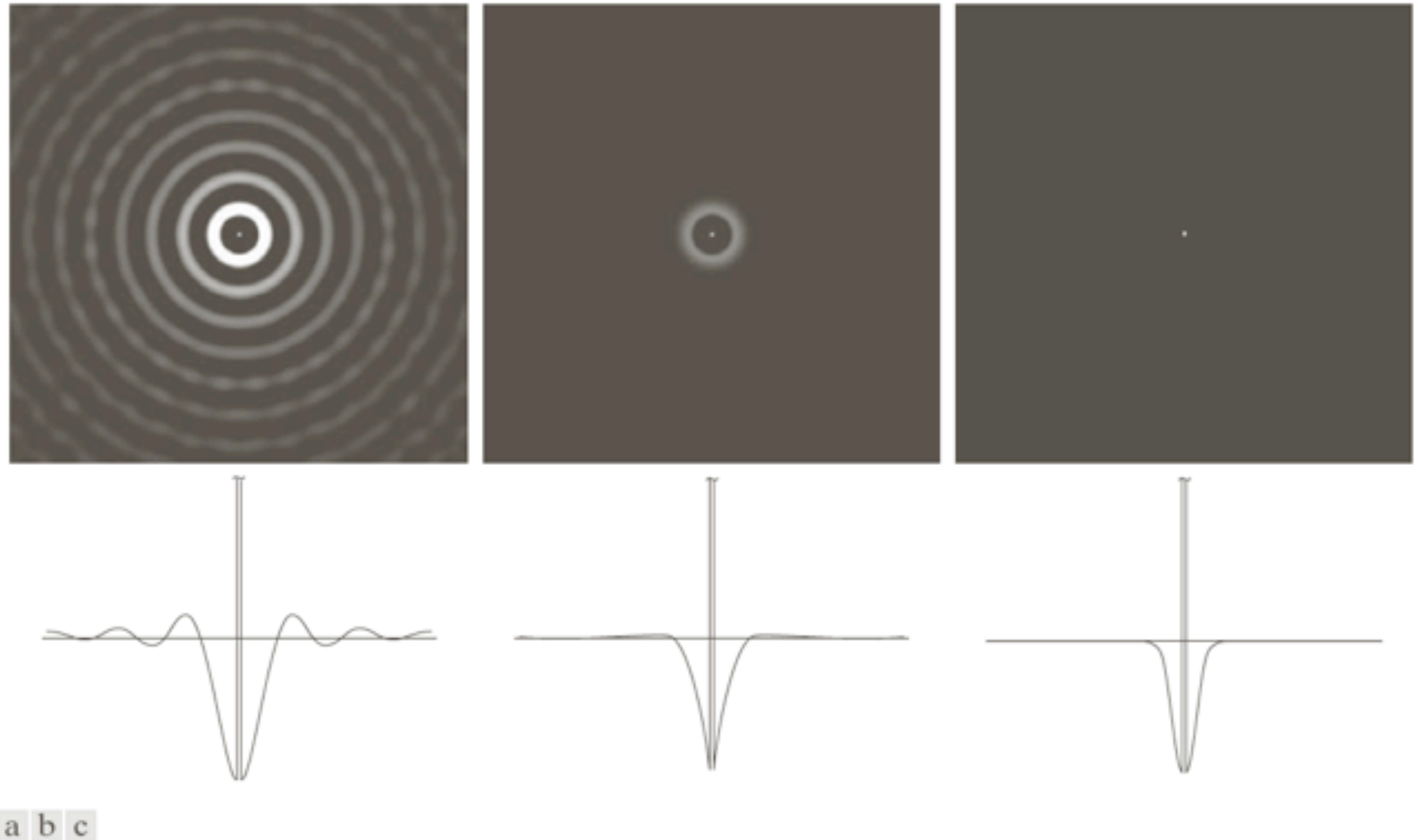


FIGURE 4.53 Spatial representation of typical (a) ideal, (b) Butterworth, and (c) Gaussian frequency domain highpass filters, and corresponding intensity profiles through their centers.

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.

a b c

FIGURE 4.55 Results of highpass filtering the image in Fig. 4.41(a) using a BHPF of order 2 with $D_0 = 30, 60$, and 160, corresponding to the circles in Fig. 4.41(b). These results are much smoother than those obtained with an IHPF.

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.

Laplacian in the frequency domain

$$H(u,v) = -4\pi^2(u^2 + v^2)$$

We need to shift the center from (0,0) to the center of the image (P/2,Q/2)

$$\begin{aligned} H(u,v) &= -4\pi^2((u - P/2)^2 + (v - Q/2)^2) \\ &= -4\pi^2 D^2(u,v) \end{aligned}$$

$$\nabla^2 f(x,y) = \mathcal{F}^{-1}[H(u,v)F(u,v)]$$

Application: $$g(x,y) = f(x,y) + c[\nabla^2 f(x,y)]$$

$$g(x,y) = f(x,y) + c\mathcal{F}^{-1}[H(u,v)F(u,v)]$$

When using IFFTs, the computed Laplacian can be much larger in magnitude than *f*. Use the scaling factor to have an appropriate scaling.

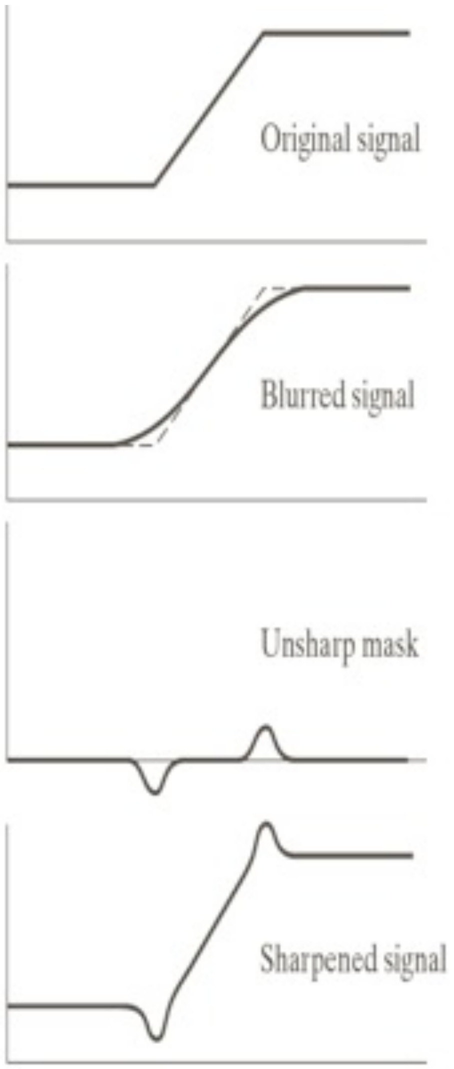
1. normalize *f* to [0,1] before DFT
2. compute the Laplacian in frequency space
3. divide the Laplacian (in space) by its maximum value (range [-1,1])

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).

Unsharp masking and highboost filtering:

corresponding operation
In frequency domain



1. Blur original image

$$f(x,y) \rightarrow \bar{f}(x,y)$$

$$f_{LP}(x,y) = \mathcal{F}^{-1}(H_{LP}(u,v)F(u,v))$$

2. Subtract blurred from image to create a mask

$$g_{\text{mask}}(x,y) = f(x,y) - \bar{f}(x,y)$$

$$g(x,y) = f(x,y) - f_{LP}(x,y)$$

3. Add the mask to the original

$$g(x,y) = f + k * g_{\text{mask}}(x,y)$$

$$\begin{aligned} g(x,y) &= \mathcal{F}^{-1}(1 + k[1 - H_{LP}(u,v)]F(u,v)) \\ &= \mathcal{F}^{-1}(1 + kH_{HP}(u,v)F(u,v)) \end{aligned}$$

$$
\begin{aligned}
g(x,y) &= \mathcal{F}^{-1}(1 + k[1 - H_{LP}(u,v)]F(u,v)) \\
&= \mathcal{F}^{-1}(1 + kH_{HP}(u,v)F(u,v))
\end{aligned}
$$

Slightly more general form

$$
g(x,y) = \mathcal{F}^{-1}(k_1 + k_2 H_{HP}(u,v)F(u,v))
$$

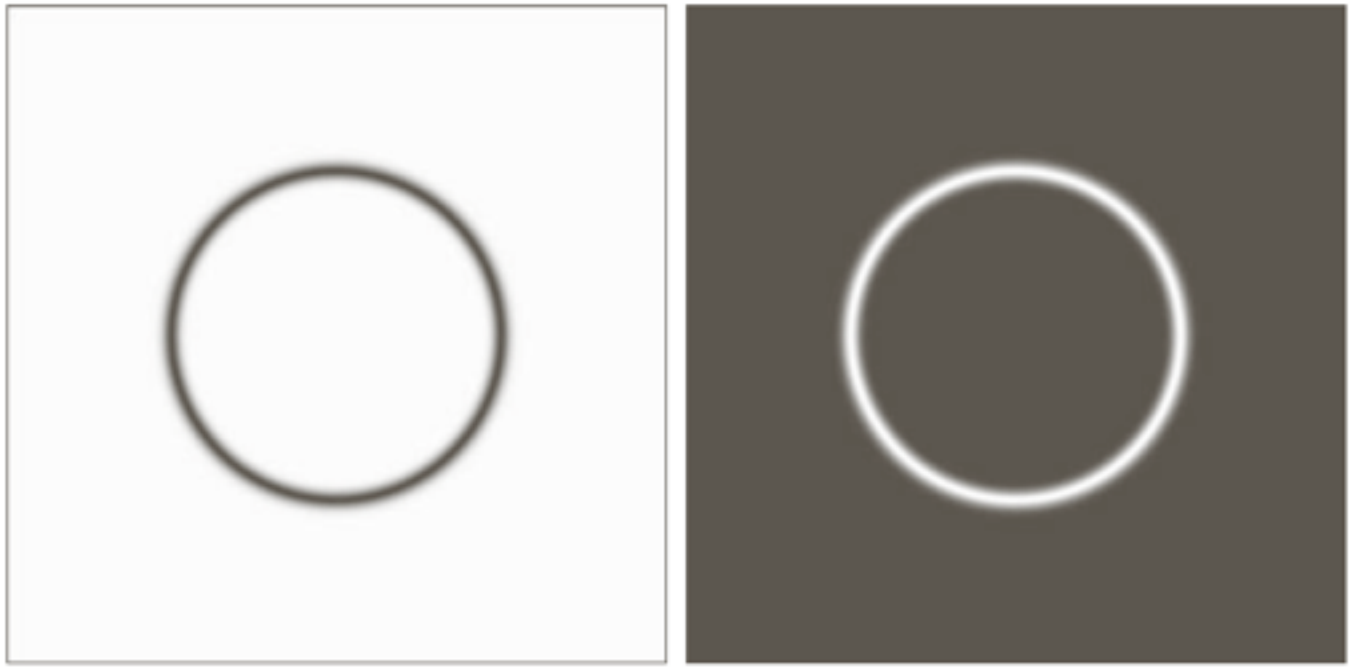
Selective filtering: band and notch

Bandpass / bandreject $\quad H_{BP}(u,v) = 1 - H_{BR}(u,v)$

Pass/reject only selected frequencies. Can be constructed using ideal/ Butterworth/Gaussian

TABLE 4.6
Bandreject filters. W is the width of the band, D is the distance $D(u, v)$ from the center of the filter, D_0 is the cutoff frequency, and n is the order of the Butterworth filter. We show D instead of $D(u, v)$ to simplify the notation in the table.

Ideal	Butterworth	Gaussian
$H(u,v) = \begin{cases} 0 & \text{if } D_0 - \frac{W}{2} \le D \le D_0 + \frac{W}{2} \\ 1 & \text{otherwise} \end{cases}$	$H(u,v) = \dfrac{1}{1 + \left[\dfrac{DW}{D^2 - D_0^2}\right]^{2n}}$	$H(u,v) = 1 - e^{-\left[\frac{D^2 - D_0^2}{DW}\right]^2}$



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.

Notch filters: pass or rejects frequencies in a pre-definite neighborhoods in a frequency rectangle.

To have zero-phase shift, the filter must be symmetric about the origin.

$$(u,v) \Longleftrightarrow (-u,-v)$$

Construction: products of HP shifted to the center of the notch

General formula:

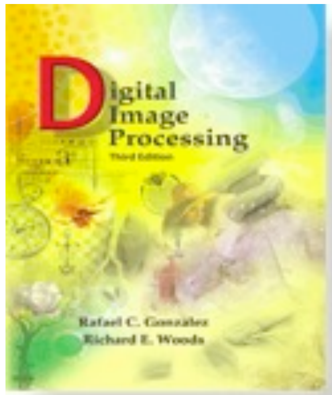
highpass filter centered at (u_k, v_k)

$$H_{NR}(u,v) = \prod_{k=1}^{Q} H_k(u,v) H_{-k}(u,v)$$

highpass filter centered at $(-u_k, -v_k)$

NR = notch reject

$$H_{NP}(u,v) = 1 - H_{NR}(u,v)$$



Chapter 4

Filtering in the Frequency Domain

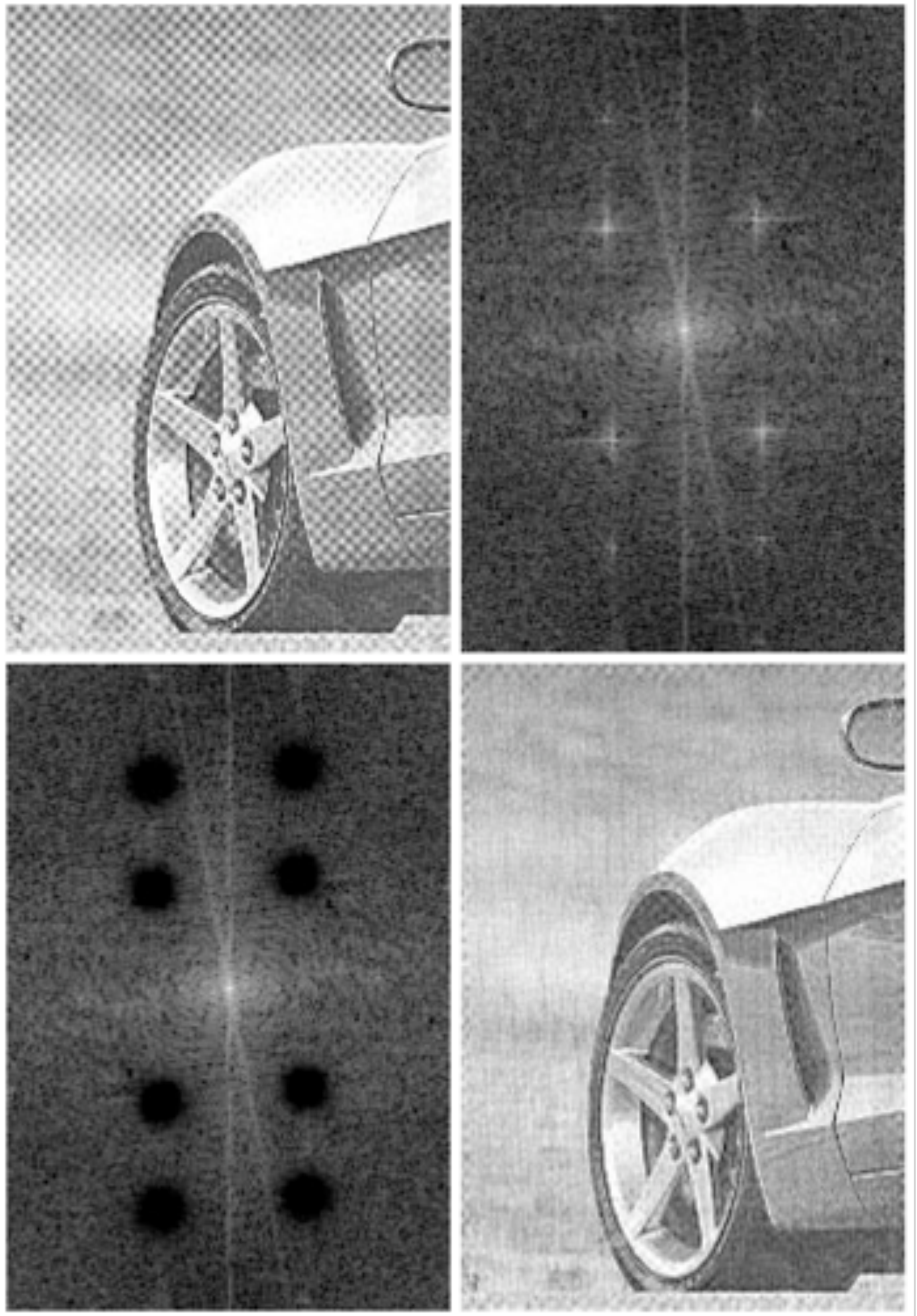
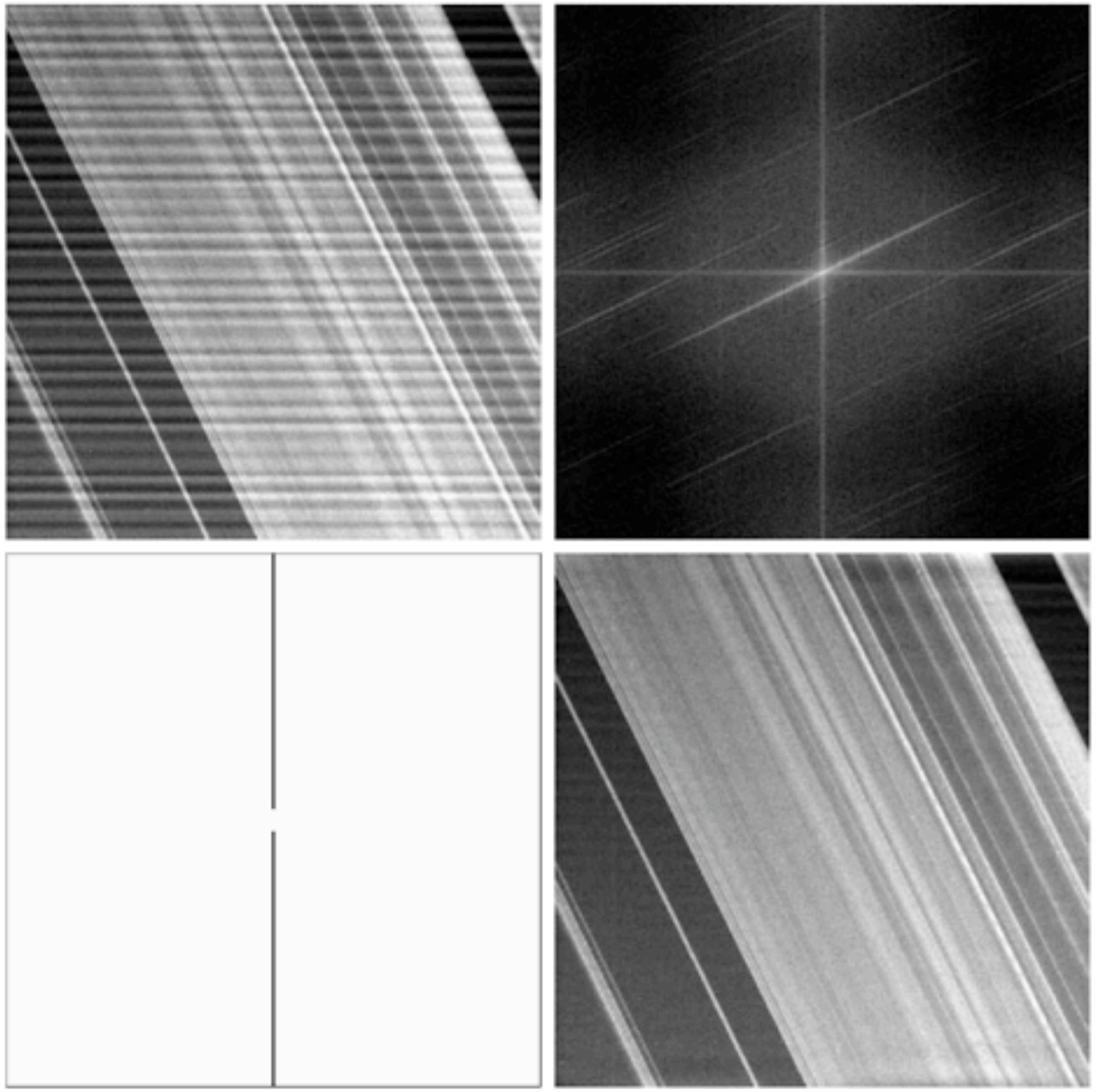
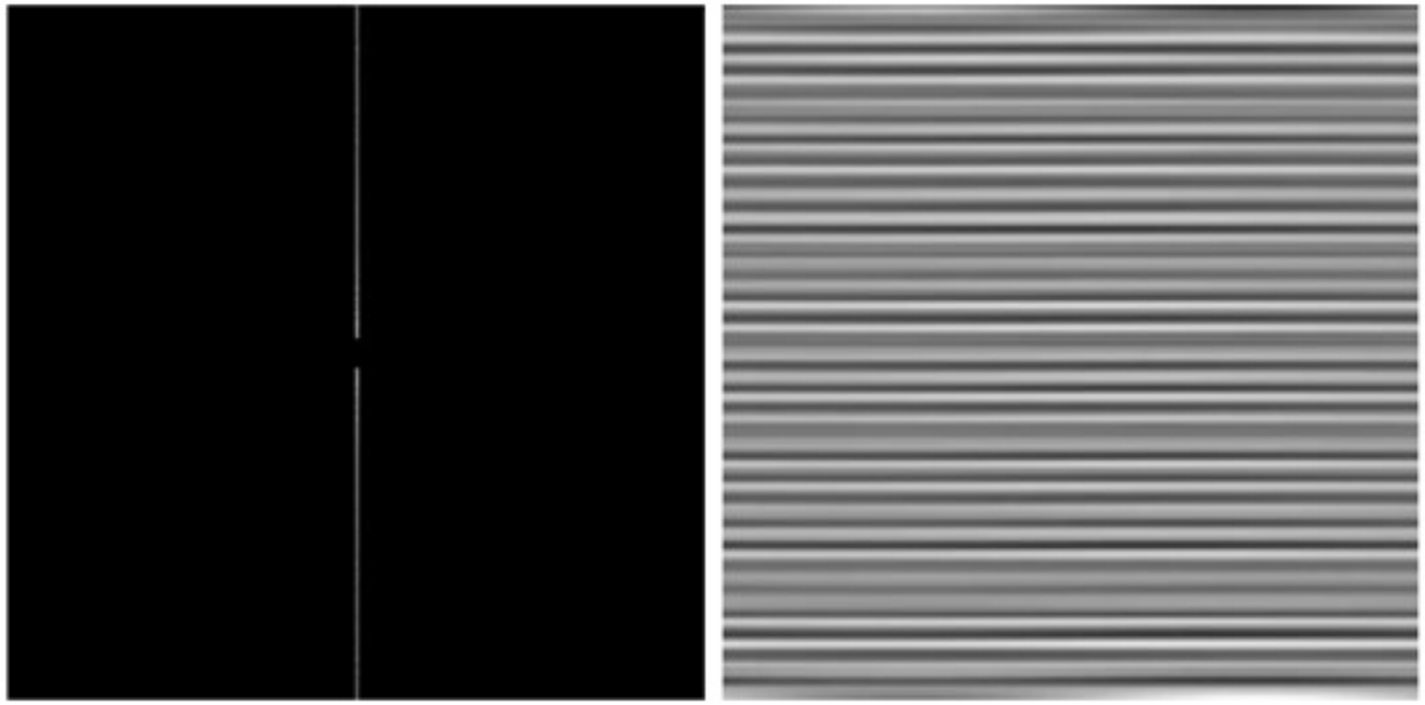


FIGURE 4.64 (a) Sampled newspaper image showing a moiré pattern. (b) Spectrum. (c) Butterworth notch reject filter multiplied by the Fourier transform. (d) Filtered image.



a b
c d

FIGURE 4.65 (a) 674 × 674 image of the Saturn rings showing nearly periodic interference. (b) Spectrum: The bursts of energy in the vertical axis near the origin correspond to the interference pattern. (c) A vertical notch reject filter. (d) Result of filtering. The thin black border in (c) was added for clarity; it is not part of the data. (Original image courtesy of Dr. Robert A. West, NASA/JPL.)



a b

FIGURE 4.66 (a) Result (spectrum) of applying a notch pass filter to the DFT of Fig. 4.65(a). (b) Spatial pattern obtained by computing the IDFT of (a).