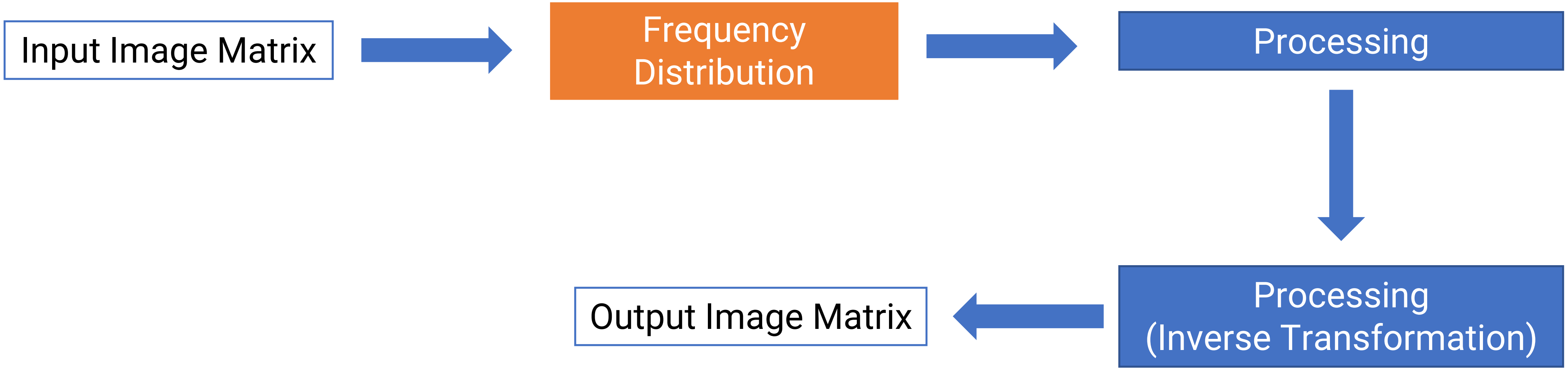
BME4120
Biomedical Image Processing

Lecture 5

Image Enhancement

Spatial Domain

Frequency Domain



Discrete Fourier Transform (DFT)

2D DFT

$$\boxed{F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi ux/M} e^{-j2\pi vy/N}}$$

$$u = 0, 1, \ldots, M-1 \quad v = 0, 1, \ldots, N-1$$

$$f(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi ux/M} e^{j2\pi vy/N}$$

Discrete Fourier Transform (DFT)

2D Inverse DFT

$$\boxed{f(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi ux/M}e^{j2\pi vy/N}}$$

$$x = 0, 1, \ldots, M-1 \qquad y = 0, 1, \ldots, N-1$$

Magnitude, Phase and Power Spectrum of FT

Magnitude:

$$|F(u,v)| = [R^2(u,v) + I^2(u,v)]^{1/2}$$

Phase:

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

Power Spectrum:

$$P(u,v) = |F(u,v)|^2 = R^2(u,v) + I^2(u,v)$$

Frequency Domain Filters

Steps to apply filters in frequency domain:

[Input image : f(x,y)]

- ❑Compute DFT of the input image (F(u,v)).
- ❑Multiply F(u,v) with DFT of the transfer function (filter function, G(u,v))
- ❑Compute inverse DFT of the result.
- ❑Obtain the real part of the DFT as the ***enhanced*** or ***filtered*** result.

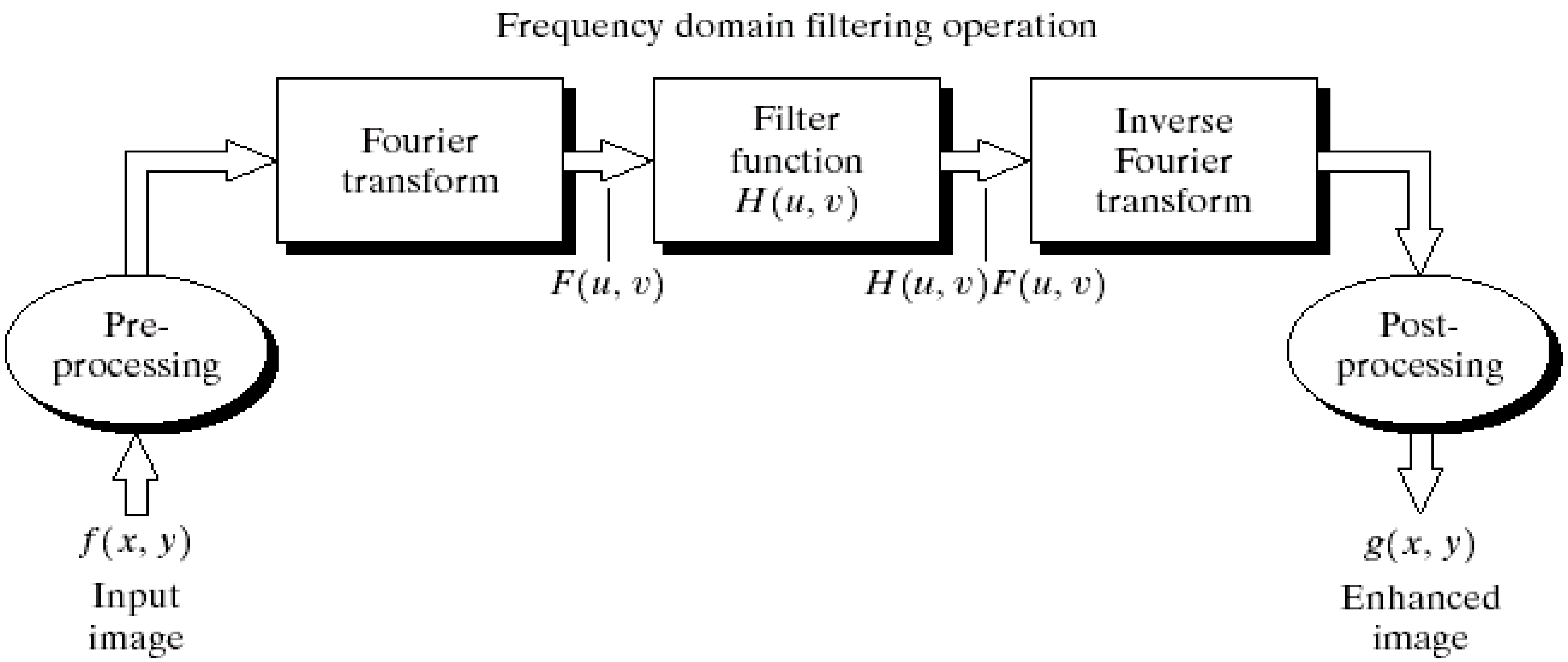


Image: Gonzalez & Woods, Digital Image Processing (2002)

Frequency Domain Filters

The frequency domain filters have some advantages as compared to the spatial domain filters:

- ❑ Convolution calculation is easier

- ❑ The periodic pattern of noise can be reduced to a limited set of high spatial frequencies → filtering becomes easy and effective.

- ❑ Computationally more efficient to implement.

- ❑ Feature extraction is easier

- ❑ Digitization makes filtering very easy and simple.
  - ➢ Fourier transform is computed as fast Fourier transform (FFT) and discrete Fourier transform (DFT) in the digital domain.

Convolution Theory

In spatial domain, filter function directly operates over the pixel values and hence, the convolution is required to compute in the time domain.

$$(f \star h)(t) = \int_{-\infty}^{\infty} f(\tau)h(t-\tau)d\tau$$

f and g are two functions in time domain

★ convolution operator

Convolution, directly in the spatial domain, is very difficult to calculate.

Can be simplified using Fourier transform.

$$\mathcal{F}(f \star g) = \mathcal{F}(f).\mathcal{F}(g) = F.G$$

Fourier transform of f

Element-by-element multiplication

Convolution in spatial domain is analogous to multiplication in frequency domain.

Convolution Theory

Inverse Fourier transform can be used to go back to spatial domain.

$$f(t) \star h(t) \Leftrightarrow \mathrm{H}(u)F(u)$$

Convolution in spatial domain is analogous to multiplication in frequency domain.

$$f(t)h(t) \Leftrightarrow \mathrm{H}(u) \star F(u)$$

Convolution in frequency domain is analogous to multiplication in spatial domain.

Frequency domain filters

Frequency domain filters are ***mainly*** divided into four different types

- ❑Smoothing domain filters or low pass filters
- ❑Sharpening domain filters or high pass filters
- ❑Homomorphic filters
- ❑Colour image enhancement methods or techniques

Smoothing Domain Filters

A.k.a. low pass filters (LPF)

→ the low pass components of digital images are retained and the high frequency components are eliminated

Sharp transitions (abrupt change in neighbour pixel intensities) in an image are the high frequency details
→ they are removed by smoothing domain filters (low pass filters)

Ideally, an LPF would retain all the low frequency components and eliminate all the high frequency components
→ practical LPFs can't do this. Blurring and ringing are common problems in the final images.

3 types of smoothing domain filters:
Ideal LPF (**ILPF**), Butterworth LPF (**BLPF**), Gaussian LPF (**GLPF**)

ILPF

Block all the frequency details, i.e., the frequencies higher than the cut-off frequency D_0, and the other terms are retained.

$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

For an image size of M x N, the distance can be measured as

$$D(u,v) = \sqrt{\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2}$$

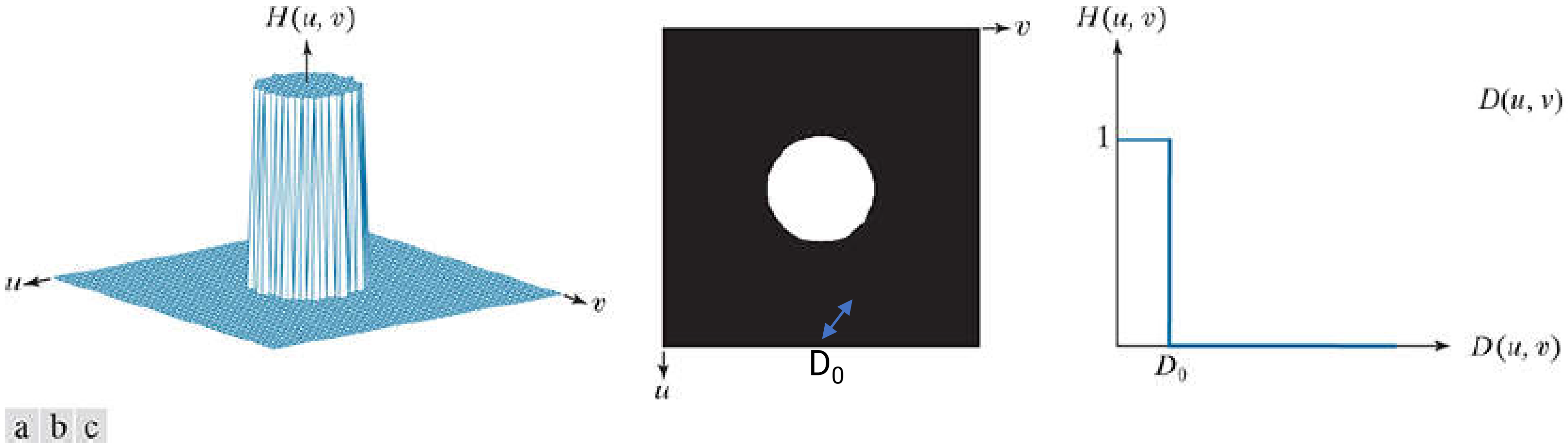


FIGURE 4.39

(a) Perspective plot of an ideal lowpass-filter transfer function. (b) Function displayed as an image. (c) Radial cross section.

D_0 determines the amount of frequency components which are passed by the filter.

As $D_0 \downarrow$, the number of image components eliminated by the filter $\uparrow$.

As $D_0 \downarrow$, the enhanced image is more blurred and ringing is more severe.

Image: Gonzalez & Woods, Digital Image Processing (2018)

Total Image Power

The lowpass filters are compared by studying their behavior as a function of the same cutoff frequencies.

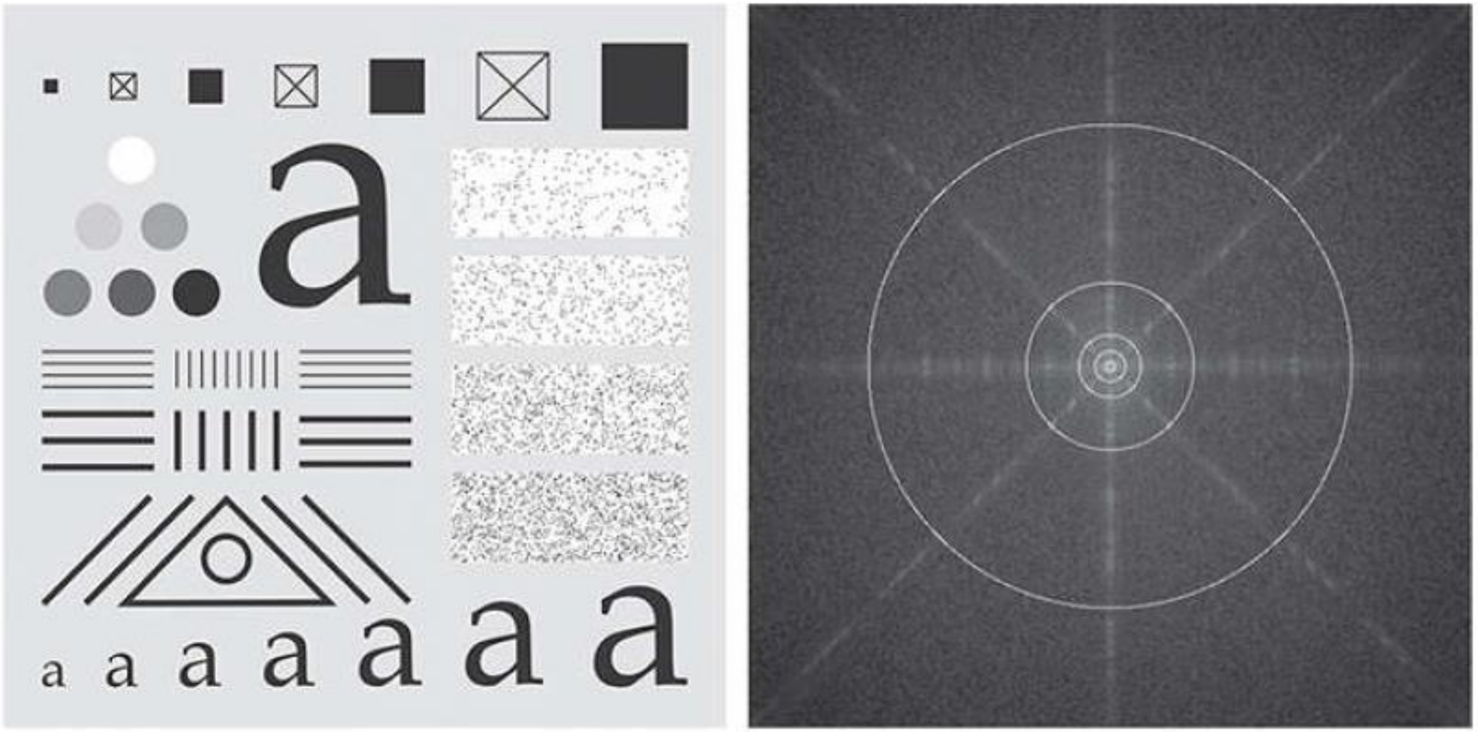
One way to establish standard cutoff frequency loci using circles that enclose specified amounts of total image power

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

If the DFT has been centered, a circle of radius with origin at the center of the frequency rectangle encloses α percent of the power, where

$$\alpha = 100\left[\sum_u \sum_v P(u,v)/P_T\right]$$

Circles for Total Image Power



a b

FIGURE 4.40
(a) Test pattern of size 688×688 pixels, and (b) its spectrum. The spectrum is double the image size as a result of padding, but is shown half size to fit. The circles have radii of 10, 30, 60, 160, and 460 pixels with respect to the full-size spectrum. The radii enclose 86.9, 92.8, 95.1, 97.6, and 99.4% of the padded image power, respectively.

Image: Gonzalez & Woods, Digital Image Processing (2018)

Circles for Total Image Power

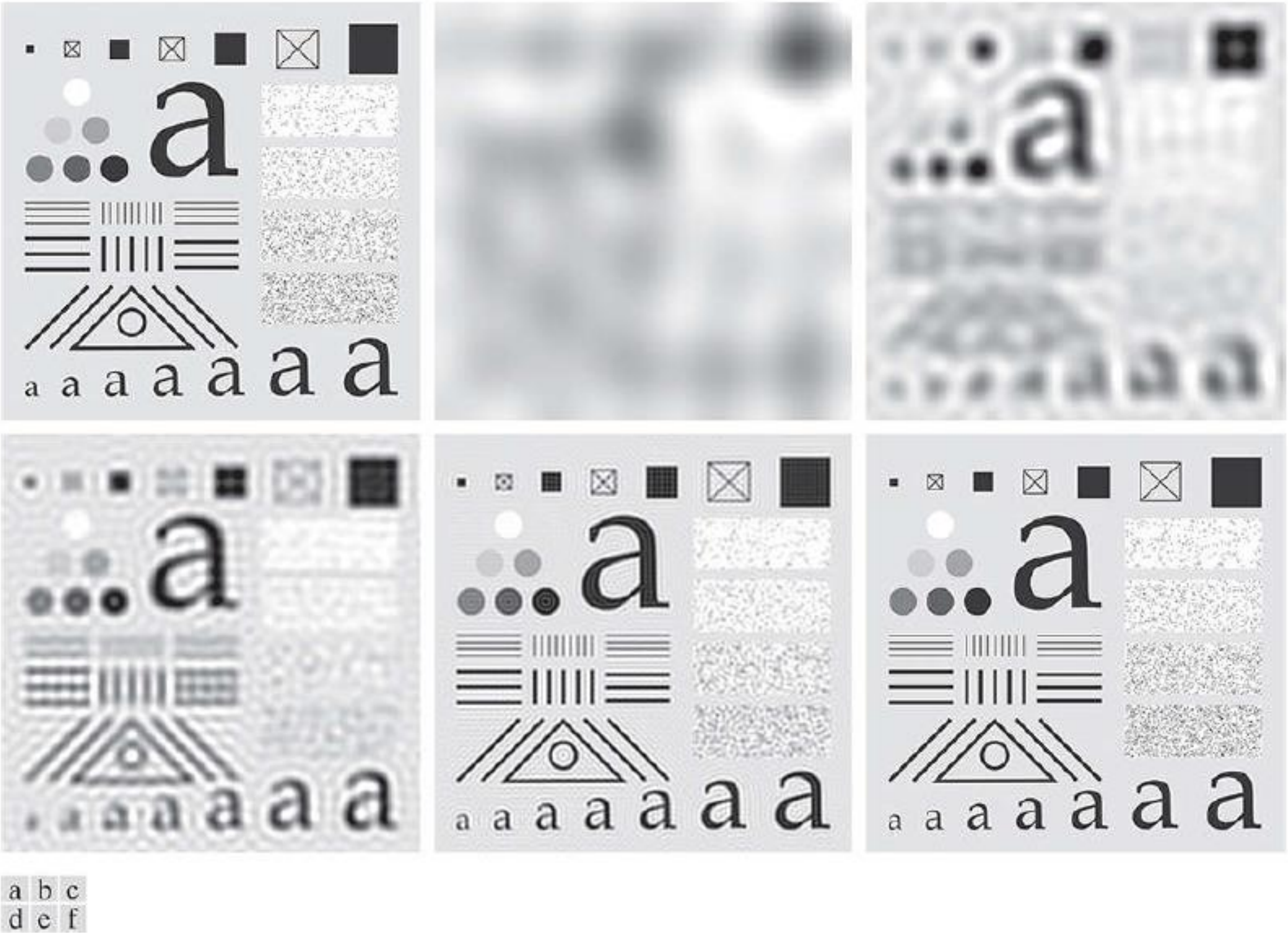


FIGURE 4.41

(a) Original image of size 688 × 688 pixels. (b)–(f) Results of filtering using ILPFs with cutoff frequencies set at radii values 10, 30, 60, 160, and 460, as shown in Fig. 4.40(b). The power removed by these filters was 13.1, 7.2, 4.9, 2.4, and 0.6% of the total, respectively. We used mirror padding to avoid the black borders characteristic of zero padding, as illustrated in Fig. 4.31(c).

Image: Gonzalez & Woods, Digital Image Processing (2018)

ILPF Weak Points

Ringing effect is the major drawback of an IPLF that occurs along the edges of the enhanced images.

The filter operation involves computation of convolution in the frequency domain and multiple peaks of the ideal low filter in the spatial domain produce ringing effect along the intensity edges in the enhanced images.

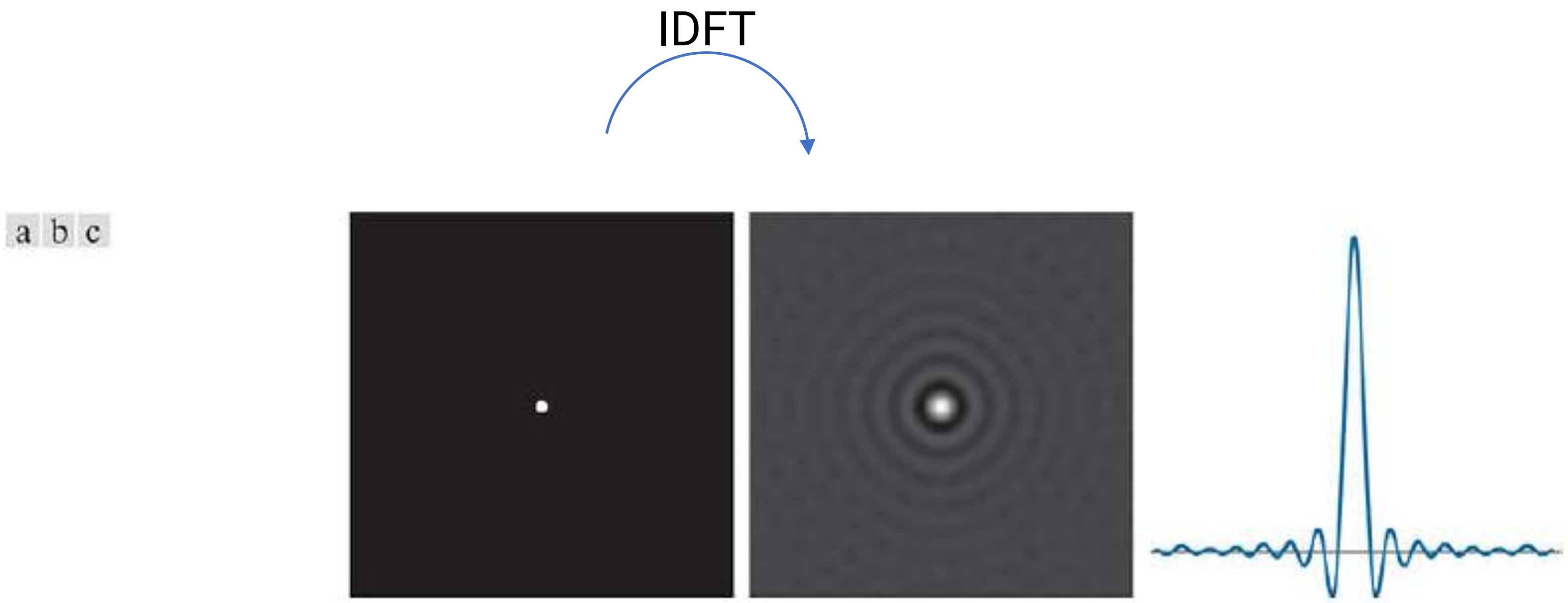


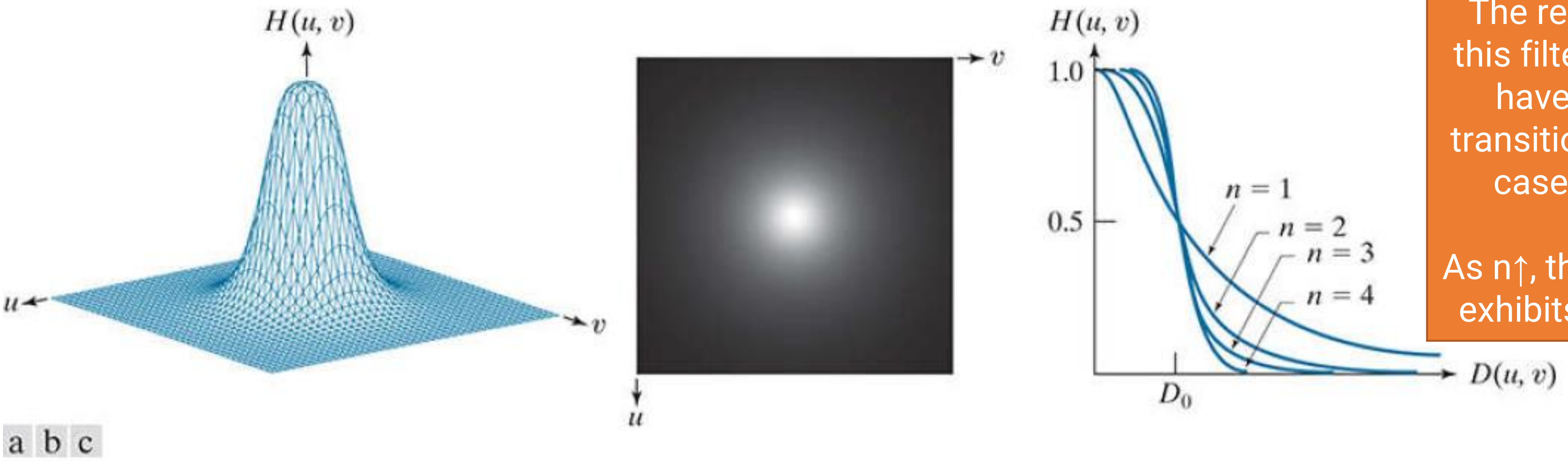
FIGURE 4.42
(a) Frequency domain ILPF transfer function. (b) Corresponding spatial domain kernel function. (c) Intensity profile of a horizontal line through the center of (b).

Image: Gonzalez & Woods, Digital Image Processing (2018)

BLPF

BLPF also eliminates the high frequency components whose transfer function of filter order n is given as

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



The response of this filter does not have a sharp transition as in the case of ILPF.

As n↑, the response exhibits like ILPF.

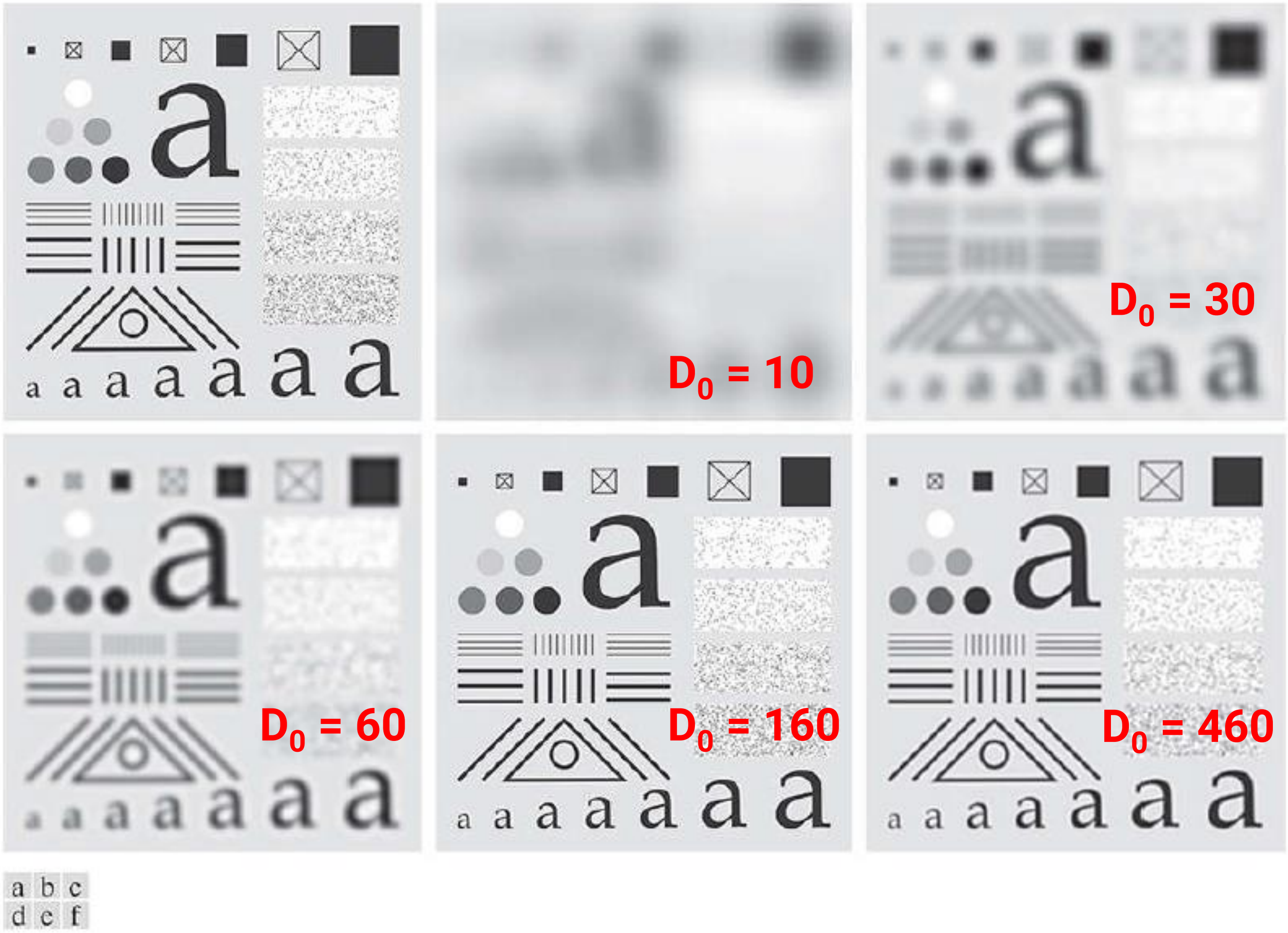
FIGURE 4.45

(a) Perspective plot of a Butterworth lowpass-filter transfer function. (b) Function displayed as an image. (c) Radial cross sections of BLPFs of orders 1 through 4.

Image: Gonzalez & Woods, Digital Image Processing (2018)

BLPF

n=2.25



BLPF has no ringing effect for small n.

Small ringing effect appears with the increase in n.

Image: Gonzalez & Woods, Digital Image Processing (2018)

GLPF

Characterized by narrow bandwidths, sharp cut-offs and low overshoots.

GLPFs are used in varied range of applications such as digital image processing, signal processing and communication systems.

Most important property: Fourier transform of a Gaussian function is also a Gaussian → the shape of filter response is the same in both the spatial and frequency domain.

$$H(u,v) = e^{-D^2(u,v)/2\sigma^2} \xRightarrow[\sigma \to D_0]{} H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

Spread
(or dispersion)

When $D(u,v) = D_0$ the GLPF transfer function is down to 0.607 of its maximum value of 1.0

GLPF

Characterized by narrow bandwidths, sharp cut-offs and low overshoots.

GLPFs are used in varied range of applications such as digital image processing, signal processing and communication systems.

Most important property: Fourier transform of a Gaussian function is also a Gaussian → the shape of filter response is the same in both the spatial and frequency domain.

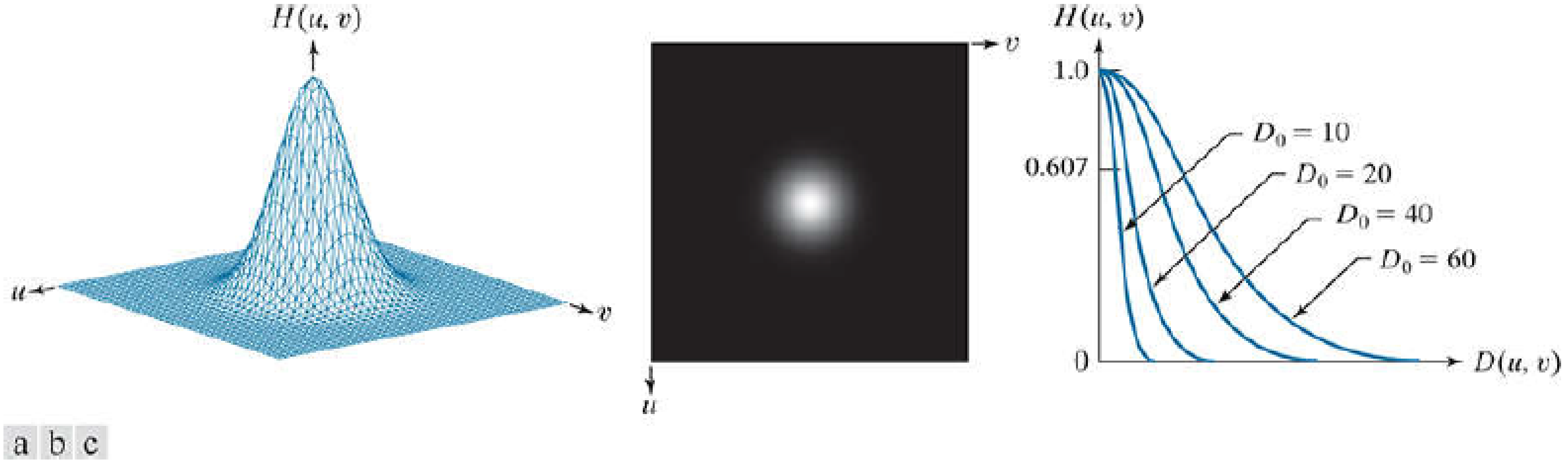
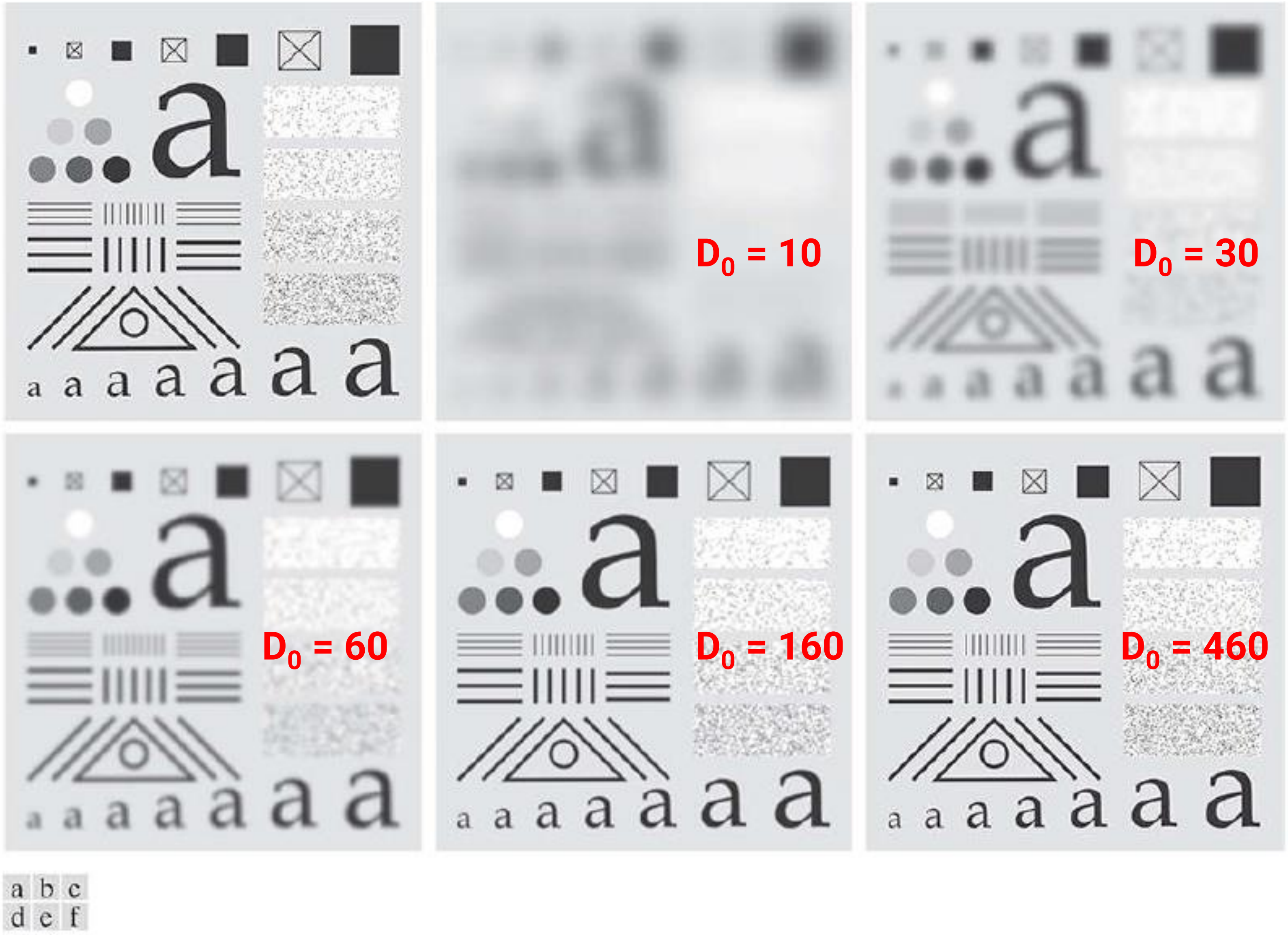


FIGURE 4.43

(a) Perspective plot of a GLPF transfer function. (b) Function displayed as an image. (c) Radial cross sections for various values of D_0.

GLPF



GPF filter has the same shape in the spatial and Fourier domain
→ no ringing effect is present in the output of this filter.

Image: Gonzalez & Woods, Digital Image Processing (2018)

ILPF, BLPF, and GLPF

Use higher values of *n* in the BLPF function → results approach the characteristics of the ILPF

Use lower values of *n* in the BLPF function → results approach the characteristics of the GLPF.

Lowpass filter transfer functions. D_0 is the cutoff frequency, and n is the order of the Butterworth filter.

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

Gonzalez & Woods, Digital Image Processing (2018)

Algorithm for smoothing domain filters

Developed based on the following factors:

Problem: Edges and sharp transitions in the gray values in an image contribute significantly to the high frequency details of its Fourier transform.

Objective: An image can be smoothed in the frequency domain by attenuating the high frequency content of its Fourier transform.

Methods: ILPF, BLPF, and GLPF are applied in the frequency domain over the image to be enhanced.

Pseudocode for Smoothing Domain Algorithm

Algorithm: Smoothing domain filtering
Input: Original Image f(x, y)
Output: Enhanced image g(x, y)

Method:
Begin
1. Acquire the input image using scanner or digital camera and convert into grayscale image f(x, y).
2. Resize and reformat the input image, if required.
3. Obtain FFT of f(x, y).
4. Compute H(u, v) of filter transfer function. For ideal, Butterworth and Gaussian low pass filters, respectively.
5. Apply H(u, v) over FFT of input image and obtain a convolution between the two.
6. Compute the inverse FFT and obtain an original image g(x, y).
7. Apply all the three smoothing filters for different values of filter order and cut-off frequency.
8. Plot the frequency response of the filter used.

End

Sharpening Domain Filters

High frequency components are contributed due to the edges, sharp transitions and other similar noise signals.

Image sharpening filters attenuate the low frequency components but the edges or high frequency details are retained.

It works in opposition to the low pass filtering and therefore, it attenuates the object and the background which does not have sharp characteristic and enhances the edges.

The random noise signals associated with the image are also enhanced by the sharpening domain or high pass filters. A transfer function as the inverse of transfer function of the low pass filter is given as

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

Sharpening Domain Filters

IHPF

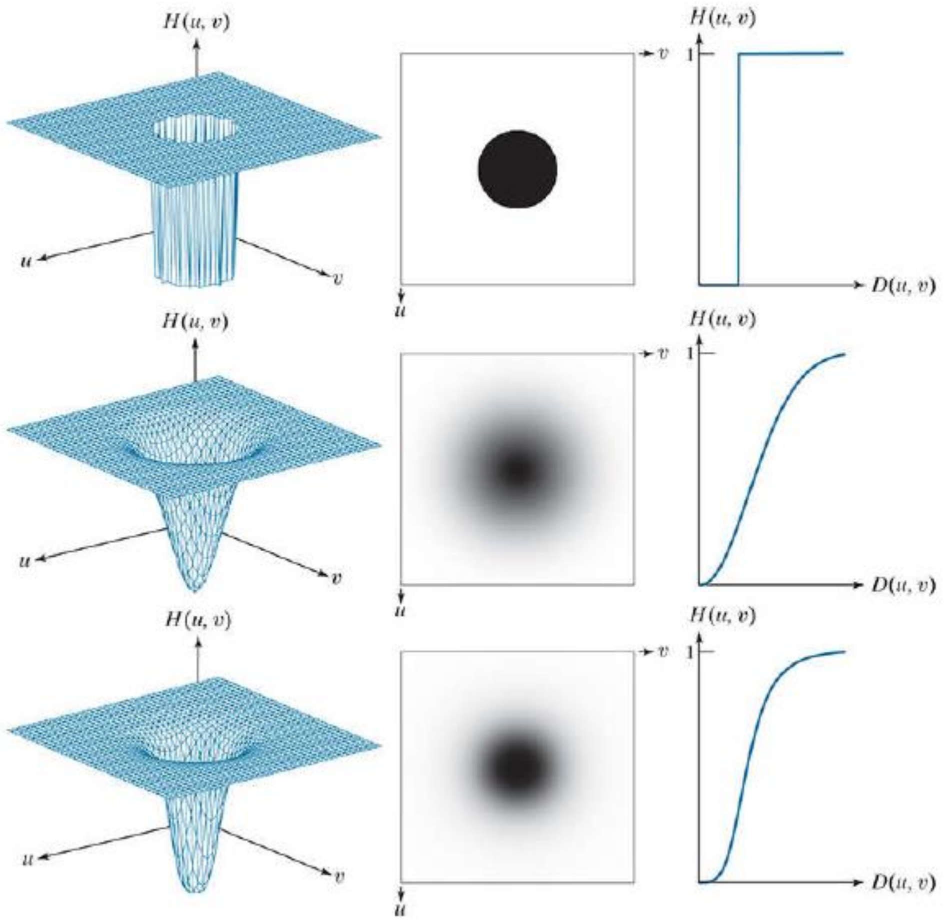
$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \leq D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

GHPF

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

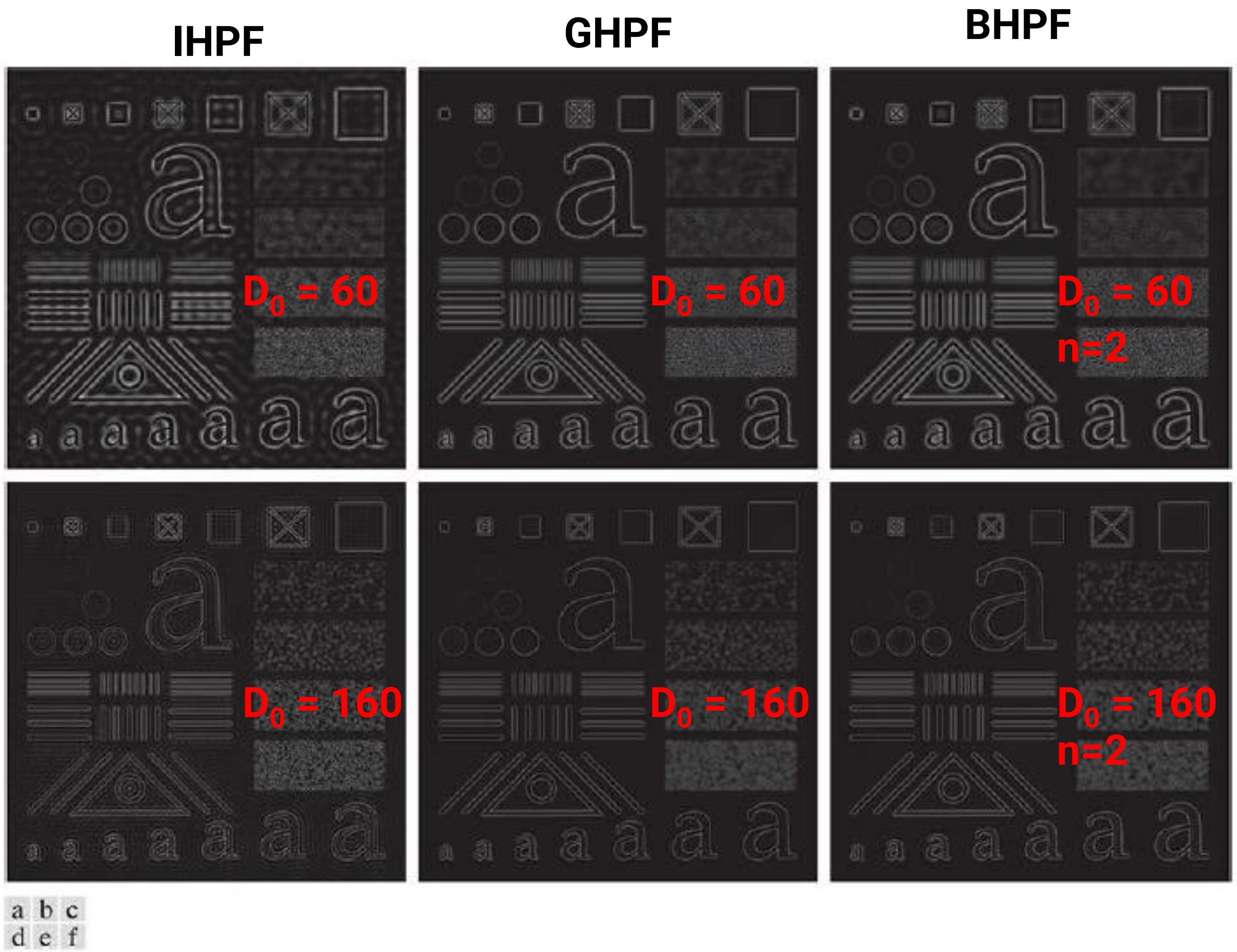
BHPF

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

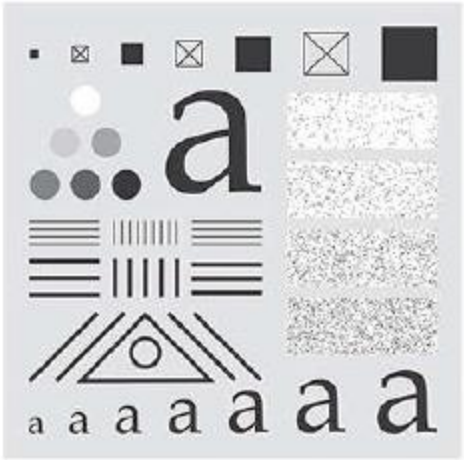


Gonzalez & Woods, Digital Image Processing (2018)

Sharpening Domain Filters



Original



Gonzalez & Woods, Digital Image Processing (2018)

Algorithm for sharpening domain filters

Developed based on the following factors:

Problem: Edges and sharp transitions in the grayscale values in an image that contribute significantly to the high frequency content are to be detected.

Objective: Low frequency contents are to be attenuated.

Methods: Ideal, Butterworth and Gaussian high pass filters are to be used in the frequency domain for image enhancement.

Pseudocode for Sharpening Domain Algorithm

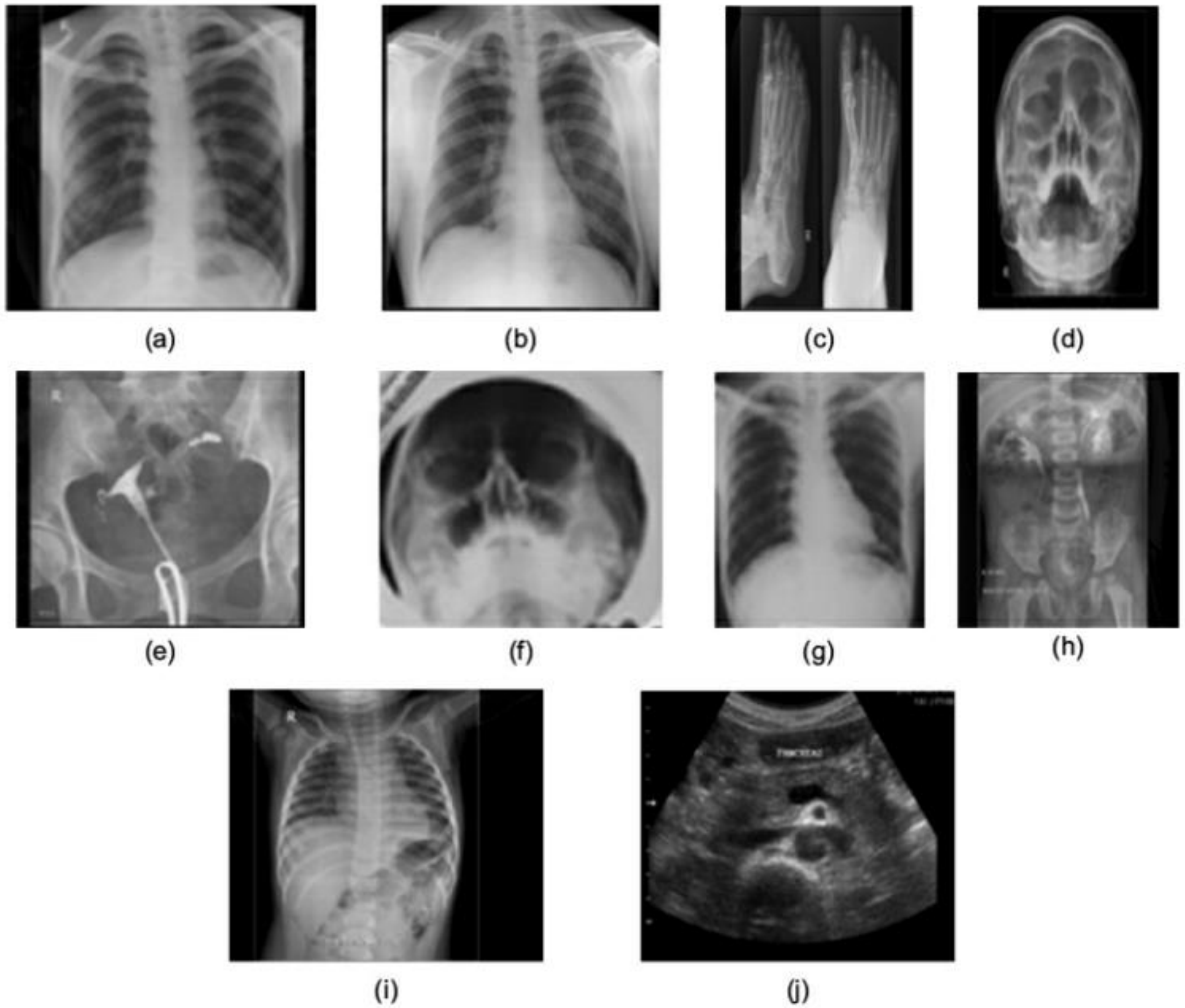
Algorithm: Sharpening domain filtering
Input: Original Image f(x, y)
Output: Enhanced image g(x, y)

Method:
Begin

1. Acquire the input image using scanner or digital camera and convert into grayscale image f(x, y).
2. Resize and reformat the input image, if required.
3. Obtain FFT of f(x, y).
4. Compute H(u, v) of filter transfer function.
5. Apply H(u, v) over FFT of input image and obtain a convolution between the two.
6. Compute the inverse FFT and obtain an original image g(x, y).
7. Apply all the three sharpening filters for different values of filter order and cut-off frequency.
8. Plot the frequency response of the filter used.

End

Practical Applications



Start with 10 X-ray images

Extract image statistical features such as peak signal to noise ratio (PSNR), contrast to noise ratio (CNR), mean, variance.

Compare different filters in terms of these features.

Patel, G.R.. Medical Image Processing: Concepts and Applications (p. 107). Prentice-Hall of India Pvt.Ltd.

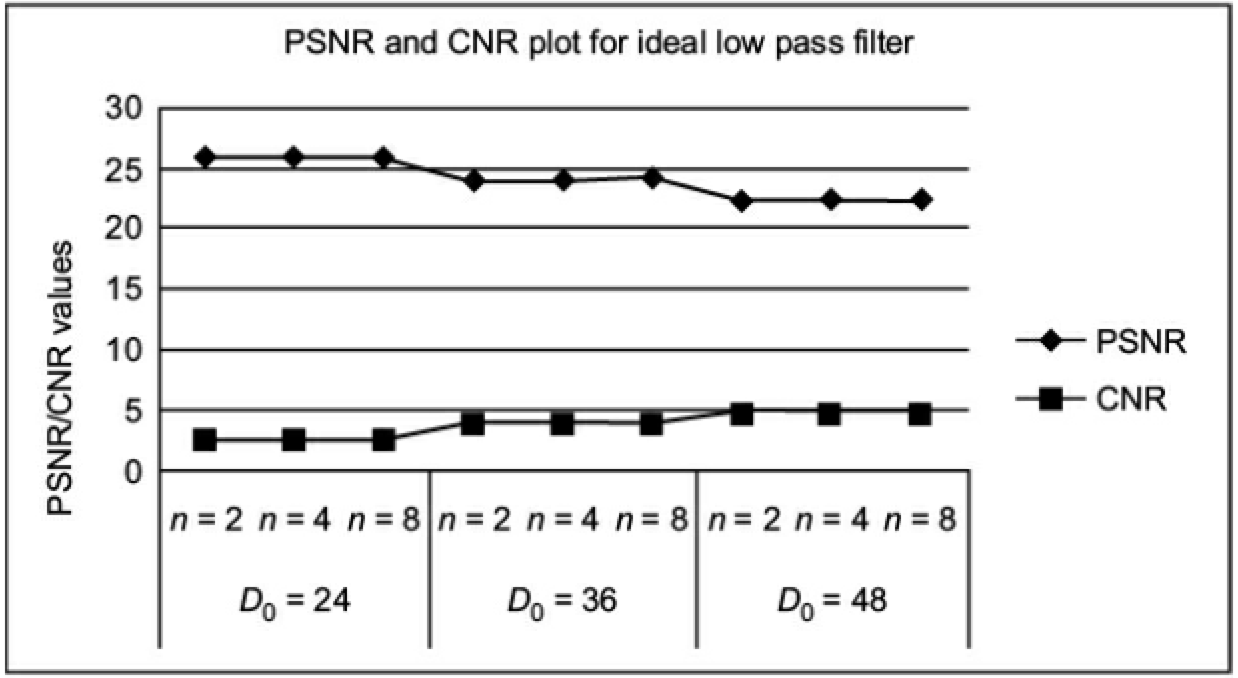
Results for Smoothing Domain Filters-1

Comparison of ILPF in Terms of PSNR and CNR
Values for Ten X-ray Images (n = 2, 4 and 8; D_0 = 24, 36 and 48)

Features \ Images			X-ray1	X-ray2	X-ray3	X-ray4	X-ray5	X-ray6	X-ray7	X-ray8	X-ray9	X-ray10
PSNR	$D_0 = 24$	$n = 2$	25.83	23.3	23.27	19.02	22.44	16.59	20.25	22.55	22.77	26.24
		$n = 4$	25.83	23.3	23.27	19.02	22.44	16.59	20.25	22.55	22.77	26.24
		$n = 8$	25.83	23.3	23.27	19.02	22.44	16.59	20.25	22.55	22.77	26.24
	$D_0 = 36$	$n = 2$	24.08	20.52	22.22	17.04	20.93	14.66	18.65	21.71	21.9	25.27
		$n = 4$	24.08	20.52	22.22	17.04	20.93	14.66	18.65	21.71	21.9	25.27
		$n = 8$	24.08	20.52	22.22	17.04	20.93	14.66	18.65	21.71	21.9	25.27
	$D_0 = 48$	$n = 2$	22.38	18.8	21.21	15.51	19.56	14	17.91	20.95	21.25	24.72
		$n = 4$	22.38	18.8	21.21	15.51	19.56	14	17.91	20.95	21.25	24.72
		$n = 8$	22.38	18.8	21.21	15.51	19.56	14	17.91	20.95	21.25	24.72
CNR	$D_0 = 24$	$n = 2$	0.25	0.003	0.005	00	0.02	0.02	0.21	0.01	0.02	0.094
		$n = 4$	0.25	0.003	0.005	00	0.02	0.02	0.21	0.01	0.02	0.094
		$n = 8$	0.25	0.003	0.005	00	0.02	0.02	0.21	0.01	0.02	0.094
	$D_0 = 36$	$n = 2$	0.38	00	0.049	0.008	0.03	0.03	0.224	0.01	0.03	0.12
		$n = 4$	0.38	00	0.049	0.008	0.03	0.03	0.224	0.01	0.03	0.12
		$n = 8$	0.38	00	0.049	0.008	0.03	0.03	0.224	0.01	0.03	0.12
	$D_0 = 48$	$n = 2$	0.47	0.003	0.049	0.008	0.04	0.04	0.27	0.01	0.04	0.12
		$n = 4$	0.47	0.003	0.049	0.008	0.04	0.04	0.27	0.01	0.04	0.12
		$n = 8$	0.47	0.003	0.049	0.008	0.04	0.04	0.27	0.01	0.04	0.12

Patel, G.R.. Medical Image Processing: Concepts and Applications (p. 107). Prentice-Hall of India Pvt.Ltd.

Results for Smoothing Domain Filters-2



The PSNR and CNR plot indicate that with the increase in cut-off distance, PSNR values are reduced and CNR values are slightly improved.

Patel, G.R.. Medical Image Processing: Concepts and Applications (p. 107). Prentice-Hall of India Pvt.Ltd.