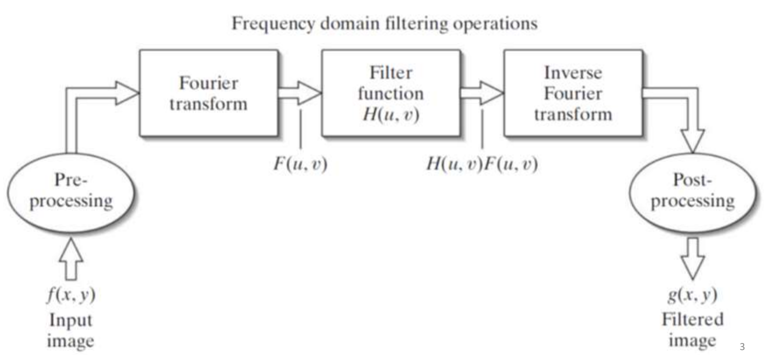
Chapter – 4

Filtering in the Frequency Domain

Introduction

- We have seen filtering in spatial domain i.e. image having domain (x, y).
- Now we will see the filtering in frequency domain (u, v).
- This is done by Fourier Transform.
- It has many applications like image enhancement, image restoration, image data compression.

Filtering in the frequency domain



Fourier Transforms

Fundamentals

The Fourier transform of functions of one variable

- The Fourier transform of a continuous function $f(t)$ of a continuous variable, t, denoted $\mathcal{F}\{f(t)\}$, is defined by the equation

$$\mathcal{F}\{f(t)\} = \int_{-\infty}^{\infty} f(t)e^{-2\pi i\mu t}dt = F(\mu)$$

- Here μ is also a continuous variable.
- t is integrated out, $\mathcal{F}\{f(t)\}$ is a function only of μ i.e. $F(\mu)$.

$$\therefore F(\mu) = \mathcal{F}\{f(t)\}$$

The inverse Fourier transform of functions of one variable

- Given $F(\mu)$, we can obtain $f(t)$ back using the inverse Fourier transform,

$$f(t) = \int_{-\infty}^{\infty} F(\mu)e^{2\pi i\mu t}d\mu$$

- The variable μ is integrated out in the inverse transform so we get simply a function of t only i.e. $f(t)$.

$$\therefore f(t) = \mathcal{F}^{-1}\{F(\mu)\}$$

- If $f(t)$ is real, its transform is complex.
- In Fourier transform, t can represent any continuous variable.
- The domain of the Fourier transform is the **frequency domain** μ.
- The units of the frequency variable μ depend on the units of t.
- If t represents time in seconds, the units of μ are cycles/sec or Hertz (Hz).
- If t represents distance in meters, then the units of μ are cycles/meter.
- The units of the frequency domain are cycles per unit of the independent variable of the input function.

The discrete Fourier transform (DFT) of one variable

- In practice, we work with a finite number of samples.
- Let $f(x)$ be continuous function having M discrete samples $f(0), f(1), \dots, f(M-1)$.
- Then the discrete Fourier transform $F(u)$ is given by

$$F(u) = \sum_{x=0}^{M-1} f(x)\, e^{-\frac{2\pi iux}{M}} \qquad u = 0, 1, \dots, M-1$$

- Here $F(0), F(1), \dots, F(M-1)$ are M complex values corresponding to the discrete Fourier transform.

The inverse discrete Fourier transform (IDFT) of one variable

- Given $F(0),\ F(1),\ \ldots,\ F(M-1)$, we can recover the samples $f(0),\ f(1), \ldots,\ f(M-1)$ using the inverse discrete Fourier transform (IDFT)

$$f(x) = \frac{1}{M}\sum_{u=0}^{M-1} F(u)e^{\frac{2\pi i u x}{M}} \qquad x = 0, 1, 2, \ldots, M-1$$

$f(x)$		$F(u)$
$\{1, 2, 3, 4\}$	$\Leftrightarrow$	$\{(10+0j),(-2+2j),(-2+0j),(-2-2j)\}$
$\{1j, 2j, 3j, 4j\}$	$\Leftrightarrow$	$\{(0+2.5j),(.5-.5j),(0-.5j),(-.5-.5j)\}$
$\{2, 1, 1, 1\}$	$\Leftrightarrow$	$\{5, 1, 1, 1\}$
$\{0, -1, 0, 1\}$	$\Leftrightarrow$	$\{(0+0j),(0+2j),(0+0j),(0-2j)\}$
$\{2j, 1j, 1j, 1j\}$	$\Leftrightarrow$	$\{5j,\ j,\ j,\ j\}$
$\{0j, -1j, 0j, 1j\}$	$\Leftrightarrow$	$\{0, -2, 0, 2\}$
$\{(4+4j), (3+2j), (0+2j), (3+2j)\}$	$\Leftrightarrow$	$\{(10+10j),(4+2j),(-2+2j),(4+2j)\}$
$\{(0+0j),(1+1j),(0+0j),(-1-j)\}$	$\Leftrightarrow$	$\{(0+0j),(2-2j),(0+0j),(-2+2j)\}$

The 2D discrete Fourier transform (DFT) of Image

- Suppose $f(x, y)$ is a digital image of size $M \times N$.
- Here $x = 0, 1, \ldots, M-1$ and $y = 0, 1, \ldots, N-1$.
- 2-D discrete Fourier transform (DFT) is given by

$$F(u, v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x, y)e^{-2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

- Here $u = 0, 1, \ldots, M-1$ and $v = 0, 1, \ldots, N-1$.

The 2D inverse discrete Fourier transform (IDFT) of Image

- Given the transform $F(u, v)$, we can obtain $f(x, y)$ by using the inverse discrete Fourier transform (IDFT)
- 2D inverse discrete Fourier transform (IDFT) is given by

$$f(x, y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u, v)e^{2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

- Here $x = 0, 1, \ldots, M-1$ and $y = 0, 1, \ldots, N-1$.

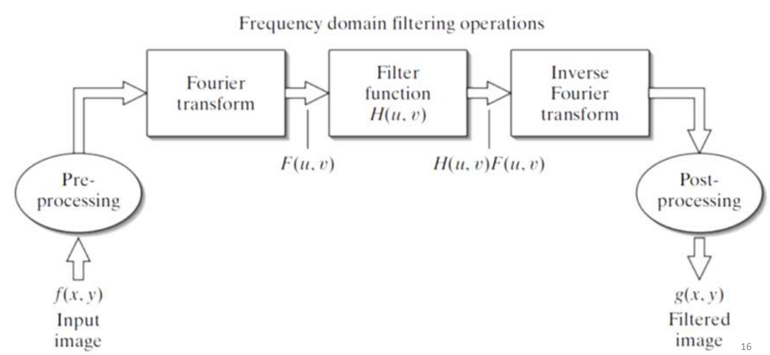
Visualizing a Fourier Transform

- $f(x, y)$ is a real function but its Fourier transform $F(u, v)$ is complex in general.
- We can analyzing a transform visually by computing its **spectrum** because it is real function.
- Spectrum can be visualized as an image.
- Let $F(u, v) = R(u, v) + i\, I(u, v)$.
- Here $R(u, v)$ is real part of $F(u, v)$ and $I(u, v)$ is imaginary part of $F(u, v)$.

- Then the Fourier spectrum of $F(u, v)$ is defined as
$$|F(u, v)| = [R^2(u, v) + I^2(u, v)]^{\frac{1}{2}}$$
- The phase angle of the transform is defined as
$$\phi(u, v) = \tan^{-1}\left(\frac{I(u, v)}{R(u, v)}\right)$$
- These two functions can be used to express the complex function $F(u, v)$ in polar form
$$F(u, v) = |F(u, v)|e^{i\phi(u,v)}$$
- The power spectrum is defined as the square of the magnitude
$$P(u, v) = |F(u, v)|^2 = R^2(u, v) + I^2(u, v)$$
- We can $P(u, v)$ or $|F(u, v)|$ to visualize Fourier transform.

Filtering in Frequency domain

Basic steps for filtering in the frequency domain



Why filtering in frequency domain?

- Filtering in the frequency domain consists of modifying the Fourier transform of an image, then computing the inverse transform to obtain the spatial domain representation of the processed result.
- Filtering in the spatial domain is more efficient computationally than frequency domain filtering when the **filters are small**.
- Filtering using an FFT algorithm can be faster than a spatial implementation when the **filters have on the order of 32** or more element.

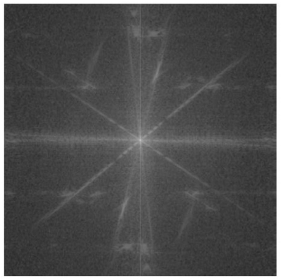
17

Example: filtering in frequency domain

- Consider the 600×600 pixel image f.

18

- Obtaining the Fourier spectrum of f, we get



19

- Result obtained in the frequency domain using the Sobel filter is as follows

20

- Considering the absolute value of output, we get

- Converting to binary image using some threshold value, we get

Original Image

Filtered Image using frequency domain

- This way we can detect edges in the image using frequency domain filtering.

Image Smoothing using Lowpass Frequency Domain Filters

- We will consider three types of lowpass filters for smoothing: ideal, Butterworth, and Gaussian.
- These three categories cover the range from very sharp (ideal) to very smooth (Gaussian) filtering.
- The shape of a Butterworth filter is controlled by a parameter called the filter order.
- For large values of this parameter, the Butterworth filter approaches the ideal filter.
- For lower values, the Butterworth filter is more like a Gaussian filter.

Ideal Lowpass Filters (ILPF)

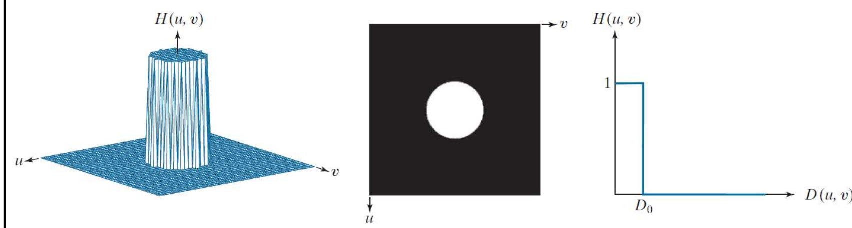
- Ideal lowpass filter (ILPF) is specified by the transfer function

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

- D_0 is a positive constant.
- $D(u,v)$ is the distance between a point (u,v) in the frequency domain and the center of $P \times Q$ frequency rectangle of padded image.

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

25

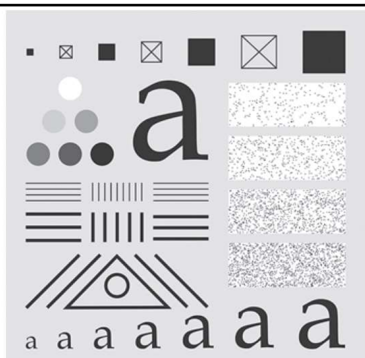


Plot of an ideal lowpass-filter transfer function

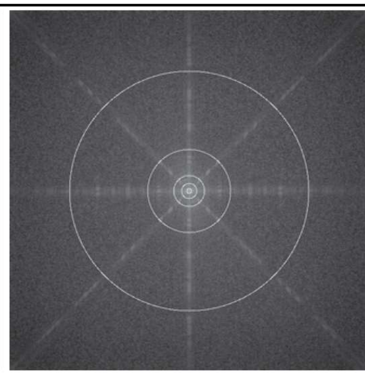
Function displayed as an image

Radial cross section

26



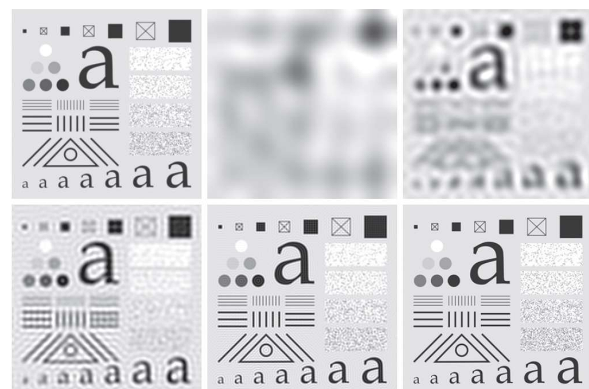
Original image



Fourier Spectrum

27

Original image



Results of filtering using **ILPFs** with D_0 = 10, 30, 60, 160, and 460

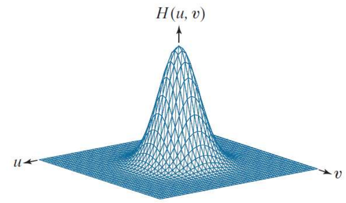
28

Gaussian Lowpass Filters (GLPF)

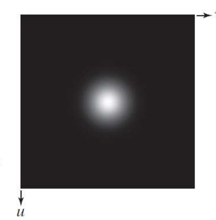
- Gaussian lowpass filter (GLPF) transfer functions have the form

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

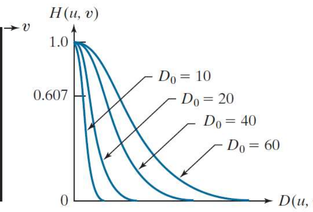
- $D(u,v)$ is the distance between a point (u,v) in the frequency domain and the center of $P \times Q$ frequency rectangle of padded image.
- D_0 is a measure of spread about the center, also called cutoff frequency.



Plot of an GLPF transfer function

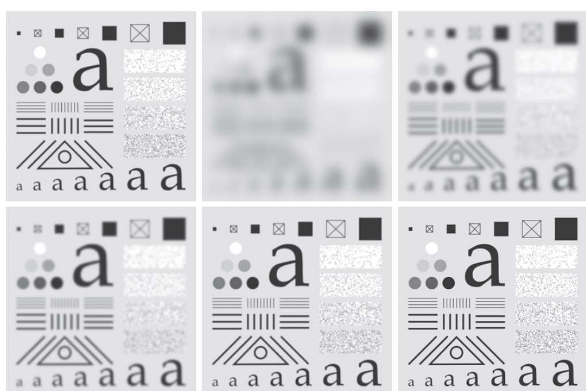


Function displayed as an image



Radial cross sections

Original image



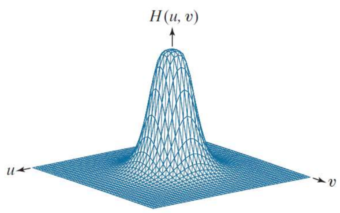
Results of filtering using **GLPFs** with D_0 = 10, 30, 60, 160, and 460

Butterworth Lowpass Filters (BLPF)

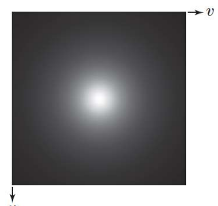
- The transfer function of a Butterworth lowpass filter (BLPF) of order n, with cutoff frequency at a distance D_0 from the center of the frequency rectangle, is defined as

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

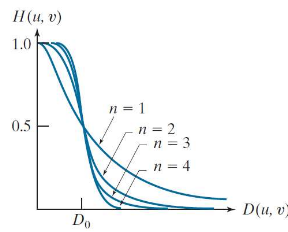
- BLPF function can be controlled to approach the characteristics of the ILPF using higher values of n, and the GLPF for lower values of n.



Plot of an BLPF transfer function

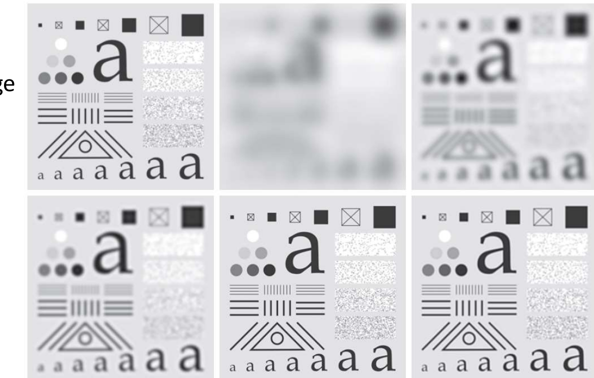


Function displayed as an image



Radial cross sections for n = 1, 2, 3, 4

Original image



Results of filtering using **BLPFs** with D_0 = 10, 30, 60, 160, and 460 and $n = 2.25$

Summary of LPF in frequency domain

Ideal

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

Gaussian

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

Butterworth

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

Image Sharpening using Highpass Filters

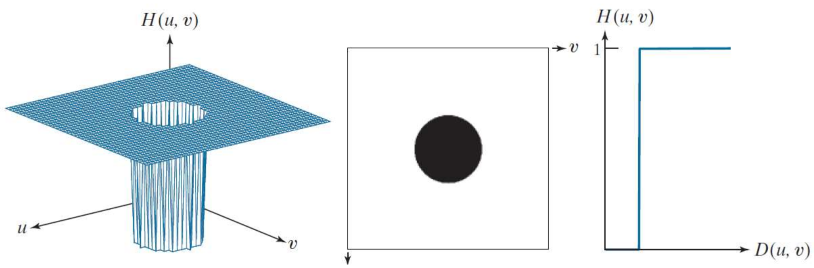
- Image sharpening can be achieved in the frequency domain by highpass filtering.
- Consider a padded Images of size $P \times Q$ and a lowpass filter transfer functions $H_{Lowpass}\,(u, v)$.
- Subtracting a lowpass filter transfer function from 1 yields the corresponding highpass filter transfer function in the frequency domain.

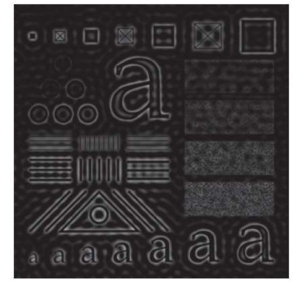
$$\therefore H_{Highpass}(u,v) = 1 - H_{Lowpass}(u,v)$$

- Here $H_{Highpass}(u,v)$ is filter transfer functions.

- Ideal highpass filter (IHPF) transfer function is given by

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

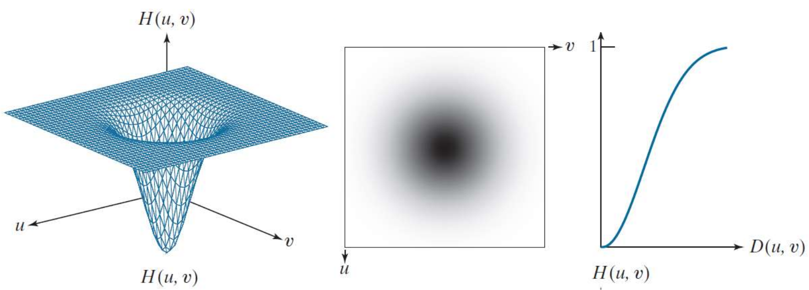


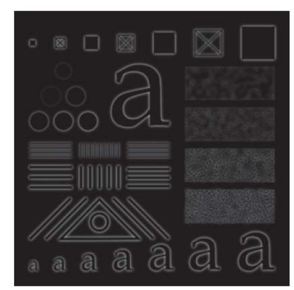


Sharpening using IHPF with $D_0 = 60$

- Gaussian highpass filter (GHPF) transfer function is given by

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

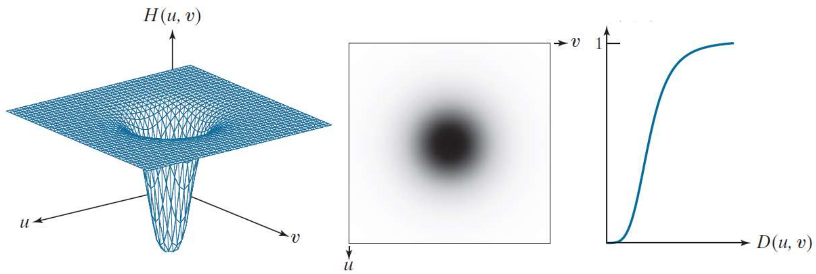


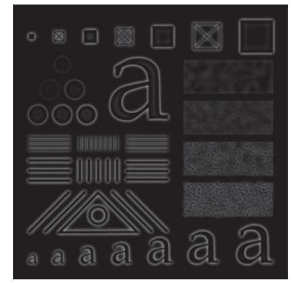


Sharpening using GHPF with $D_0 = 60$

- Butterwort highpass filter (BHPF) transfer function is given by

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





Sharpening using BHPF with $D_0 = 60$ and $n = 2$

Sharpening various HPF with $D_0 = 160$

IHPF

GHPF

BHPF ($n = 2$)

Summary of HPF in frequency domain

Ideal

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

Gaussian

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

Butterworth

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

LPF

Ideal

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

Gaussian

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

Butterworth

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

HPF

Ideal

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

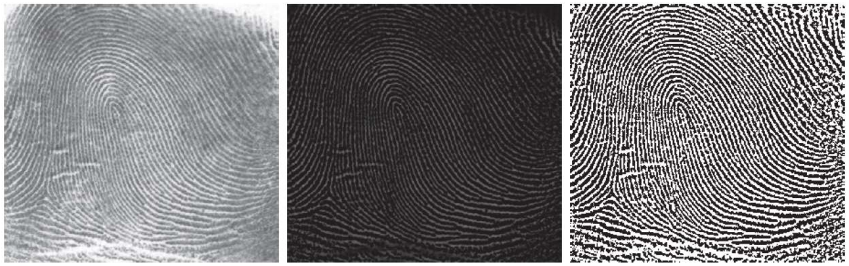
Gaussian

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

Butterworth

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

Application of filtering in frequency domain (Image enhancement)



Thumbprint Highpass filtering Thresholding

46

The Laplacian in the Frequency Domain

- We used the Laplacian for image sharpening in the spatial domain.
- It can also be used in frequency domain techniques.
- In the frequency domain filtering, following filter transfer function is used

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

Or

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

47

The Laplacian in the Frequency Domain

- Steps:
  1. Given image $f(x,y)$
  2. Find DFT of $f(x,y)$ i.e. $F(u,v) = \mathcal{F}[f(x,y)]$
  3. Multiply $F(u,v)$ by $H(u,v)$ i.e. $F(u,v)\cdot H(u,v)$
  4. Take IDFT of $F(u,v)\cdot H(u,v)$ i.e. $\mathcal{F}^{-1}[F(u,v)\cdot H(u,v)]$
- So we get

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

- Finally we get output image

$$g(x,y) = f(x,y) + c\nabla^2 f(x,y)$$

48

Original, blurry image

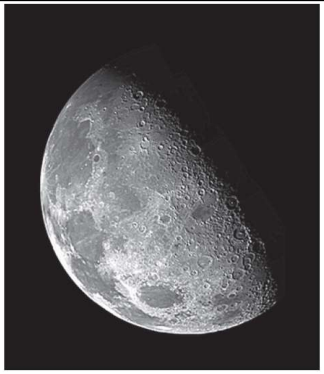


Image enhanced using the Laplacian in the frequency domain

49