An Introduction to Image Enhancement Techniques

by

Dr. Amitava Chatterjee
Electrical Measurement and Instrumentation Laboratory,
Electrical Engineering Department,
Jadavpur University, Kolkata, India.

Image Enhancement Techniques

- ✓ **The principal objective of image enhancement is to process an image so that the result is more suitable than the original image for a particular application.**

Image Enhancement Techniques

Spatial Domain Approach

Spatial domain refers to the image plane itself and involves direct manipulation of the pixels of an image

Frequency Domain Approach

Frequency domain processing techniques are based on modifying the Fourier Transform of an image

Image Enhancement Techniques

Spatial Domain Approaches

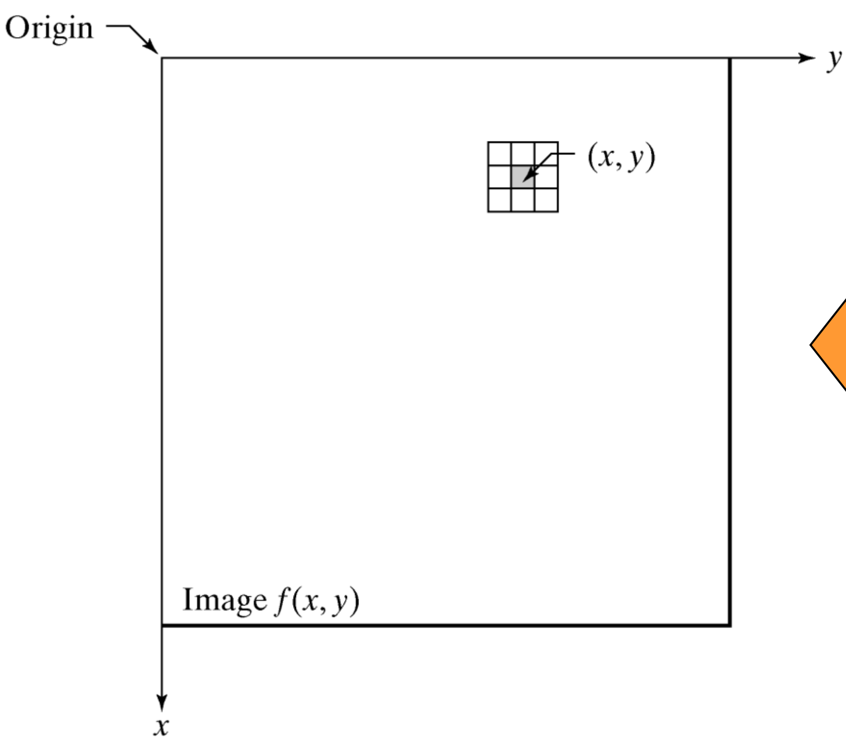
Image Processing functions in the spatial domain:

$$g(x, y) = T[f(x, y)]$$

f(*x*,*y*): The input image

g(*x*,*y*): The processed output image

T: an operation on *f* defined over some neighborhood of (*x*,*y*)



A 3 × 3 neighborhood about a point (*x*,*y*) in an image.

✓ ***Note*: The smallest neighborhood size is 1 × 1.**

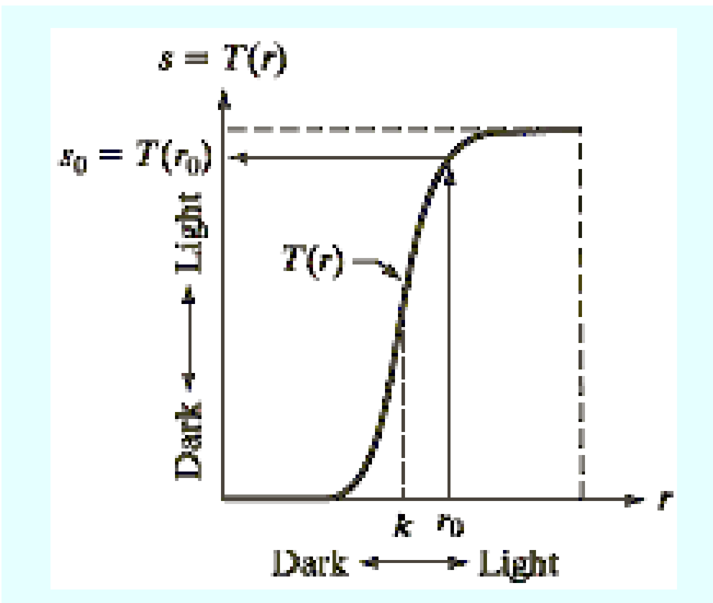
Image Enhancement Techniques

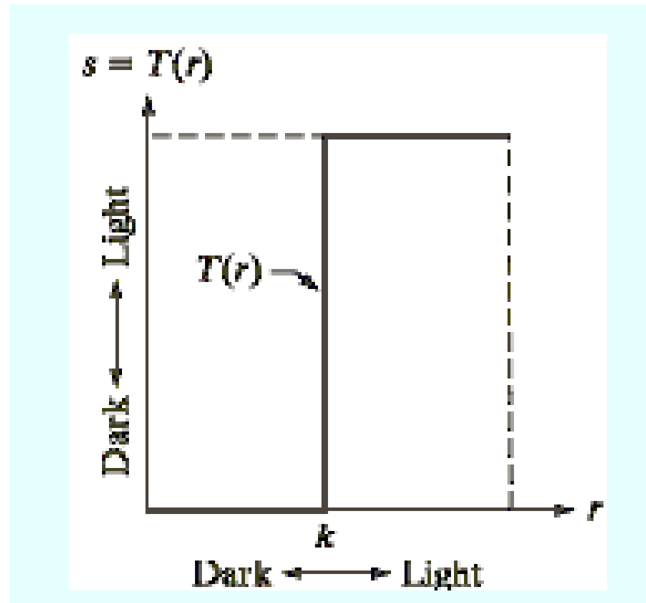
Spatial Domain Approaches

For a neighborhood size of 1 × 1 :

$$s = T(r)$$

- r: gray level of $f(x,y)$ at (x,y)
- s: gray level of $g(x,y)$ at (x,y)
- T: a gray level transformation function

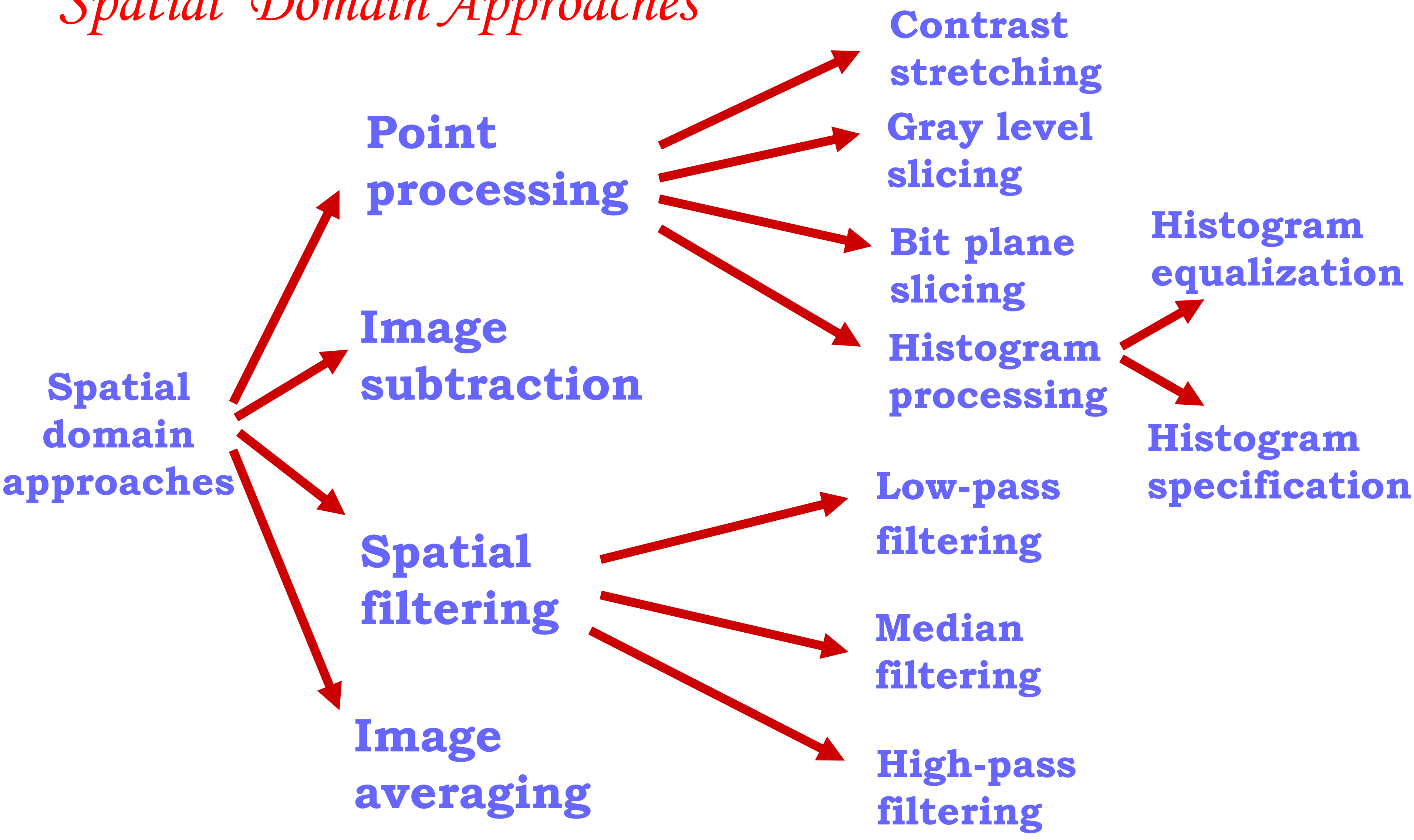




Gray level transformation functions for Contrast Enhancement.

Image Enhancement Techniques

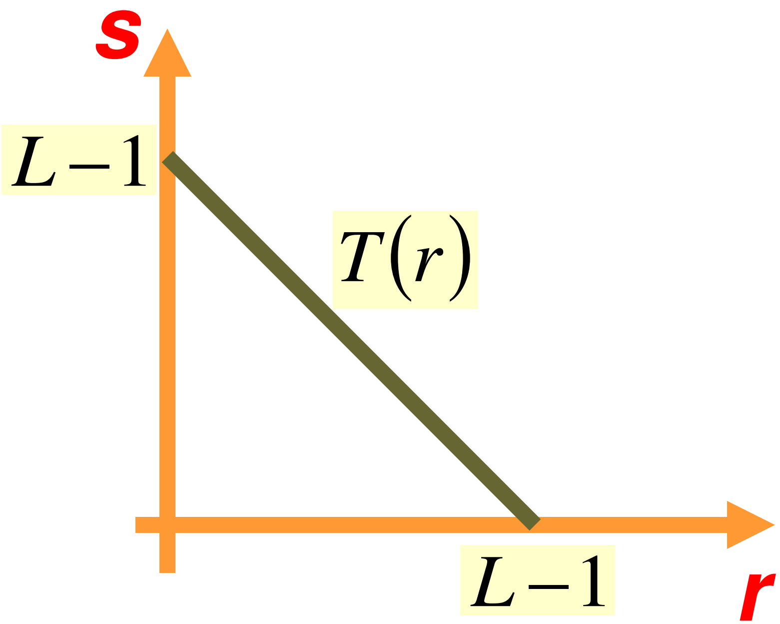
Spatial Domain Approaches



Spatial Domain Techniques

Point Processing Techniques

Negative of an Image



$s = L - 1 - r$, **for gray levels in the range** $[0, L - 1]$

Original Pepper Image

Negative of the Pepper Image

Spatial Domain Techniques

Point Processing Techniques

Contrast Stretching

- ✓ **The possible causes of a low contrast image are:**
  - **Poor illumination**
  - **Lack of dynamic range in the imaging sensor**
  - **Wrong setting of the lens aperture during image acquisition**

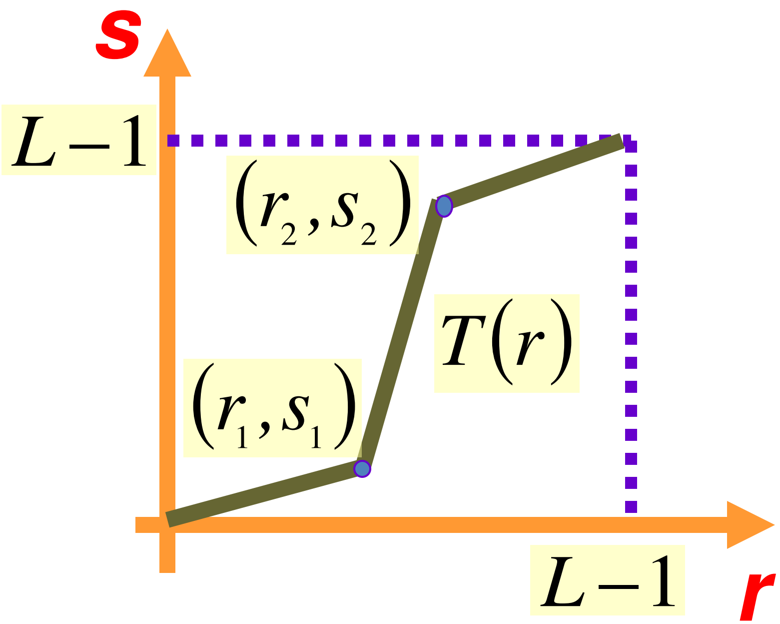
Solution ??

- ✓ **Contrast stretching attempts to increase the dynamic range of the gray levels of the image being processed.**

Spatial Domain Techniques

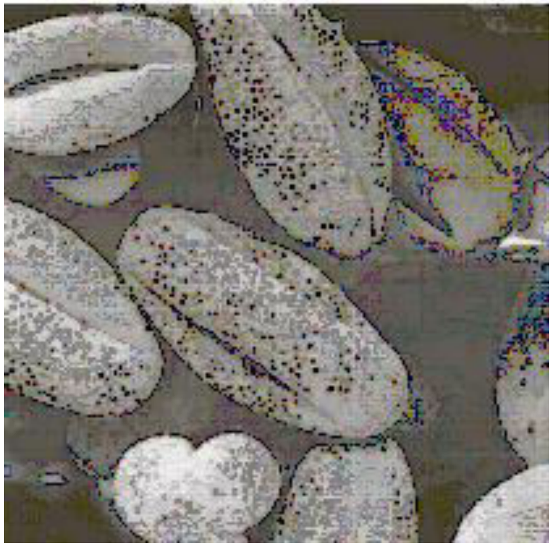
Point Processing Techniques

Contrast Stretching



- ✓ **The locations of the points (r_1, s_1) and (r_2, s_2) control the shape of the transformation function.**

A low Contrast Image



Enhanced Image after Contrast Stretching

Spatial Domain Techniques

Point Processing Techniques

Dynamic Range Compression

- ✓ **Sometimes dynamic range of a processed image far exceeds the capability of the display device.**

- ✓ **This results in only the brightest parts of the image being visible on the display screen.**

Solution ??

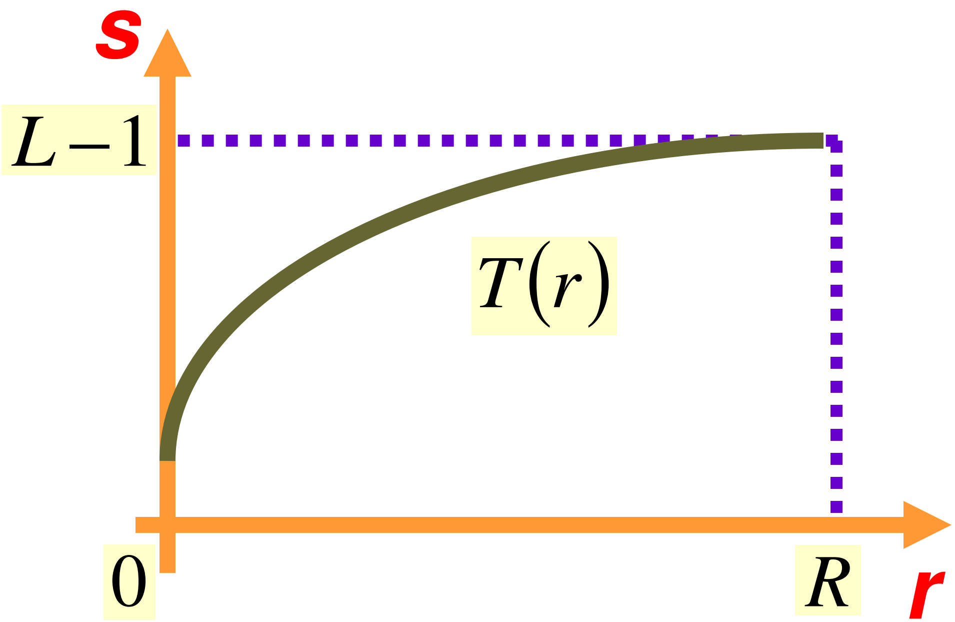
- ✓ **Compress the dynamic range of the pixel values by using an intensity transformation function:** $s = c\,log_{10}\left(1+|r|\right)$.

  Here c = a scaling constant.

Spatial Domain Techniques

Point Processing Techniques

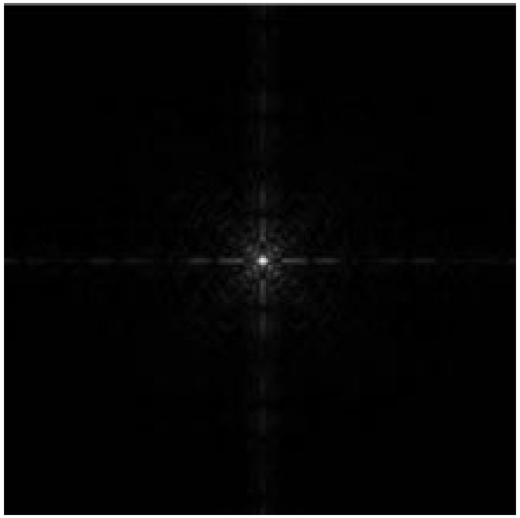
Dynamic Range Compression



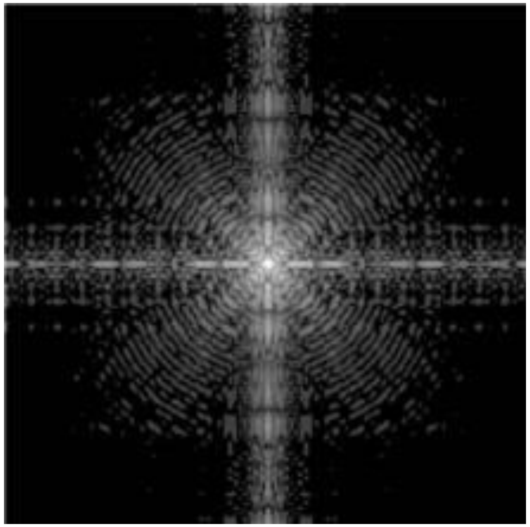
- ✓ ***Example:*** For [0, *R*] = [0, 2.5×10⁶],

$$log_{10}\left(1+|r|\right)\in\left[0, 6.4\right].$$

To scale this range up to [0, *L*-1] = [0, 255], scaling factor *c* = (255/6.4).



A Fourier spectrum, [0, *R*] = [0, 1.5 × 10⁶], scaled linearly for display in 8-bit system.

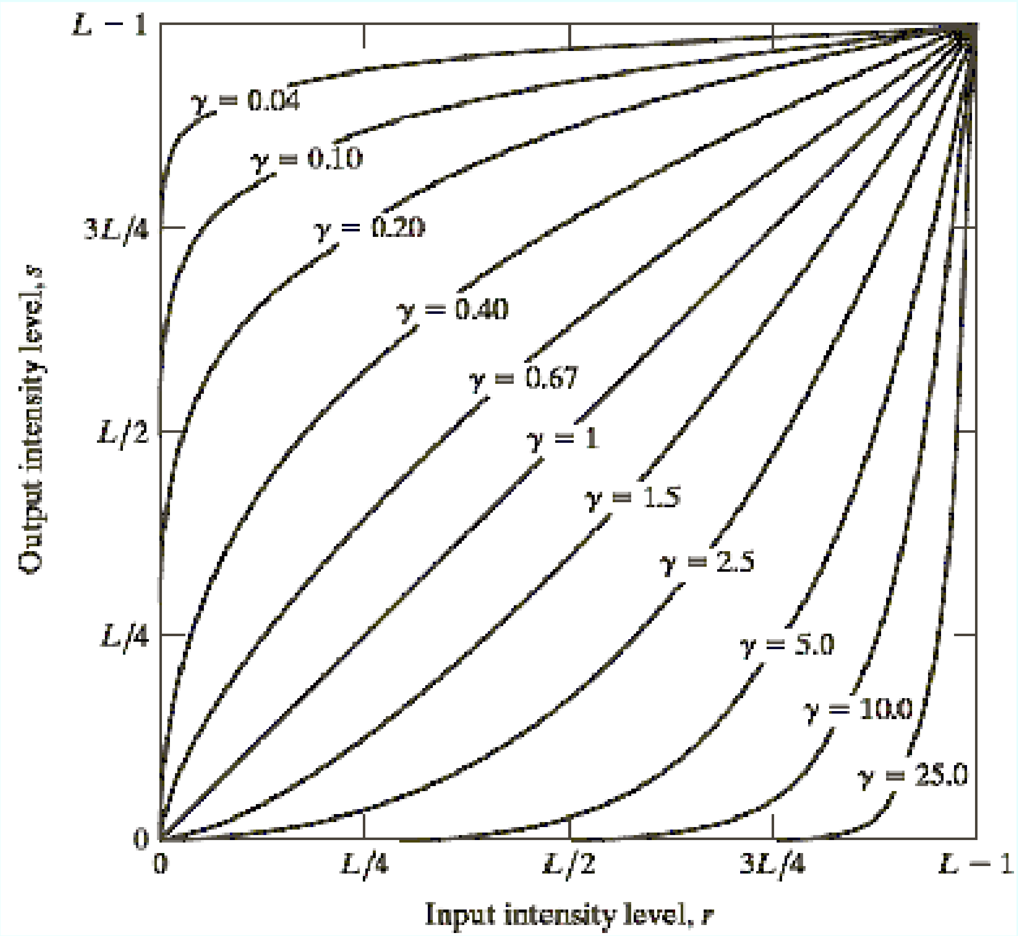


Log transformed image.

Spatial Domain Techniques

Point Processing Techniques

Power-Law (Gamma) Transformations



- ✓ ***Note*: This method is popularly known as *gamma correction* method.**

$s = cr^{\gamma}$, **with c and γ positive constants.**

Spatial Domain Techniques

Contrast Enhancement using Power-Law Transformations

MRI of a fractured human spine.

Power-Law transformed image with $c = 1$ and $\gamma = 0.6$.

Power-Law transformed image with $c = 1$ and $\gamma = 0.3$.

Spatial Domain Techniques

Point Processing Techniques

Gray-level Slicing

- ✓ Sometimes we need to highlight a specific range of gray levels in an image.
- ✓ Possible application areas are masses of water in satellite imagery, enhancement of flaws in x-ray images etc.

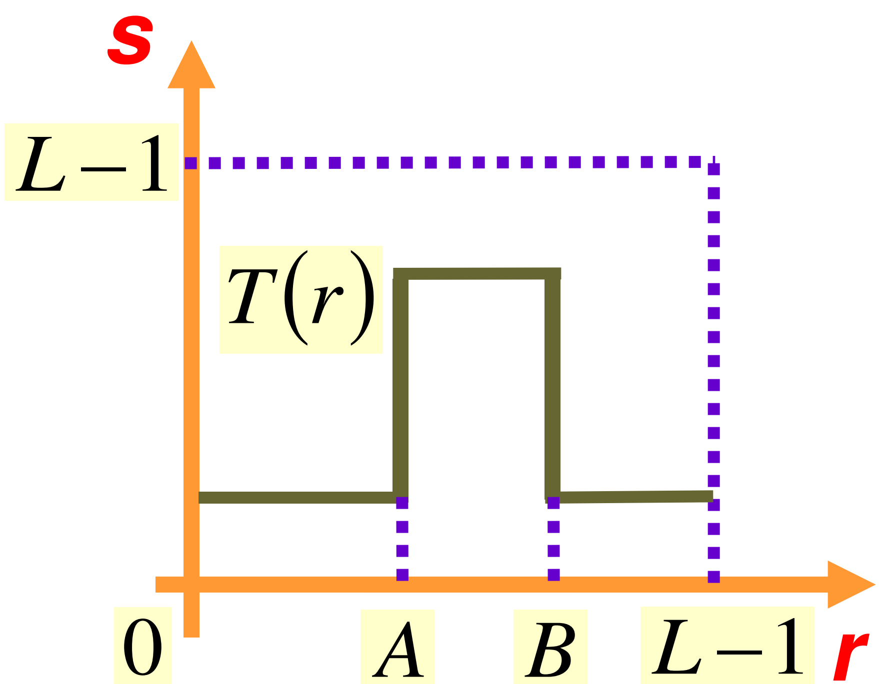
Solution ??

- ✓ Use one of the two basic approaches of gray-level slicing:
    - *Approach 1* – All gray levels in the range of interest are displayed using a high value and the rest using a low value.
    - *Approach 2* – Brightens the desired range of gray levels but preserves the background and gray-level tonalities in an image.

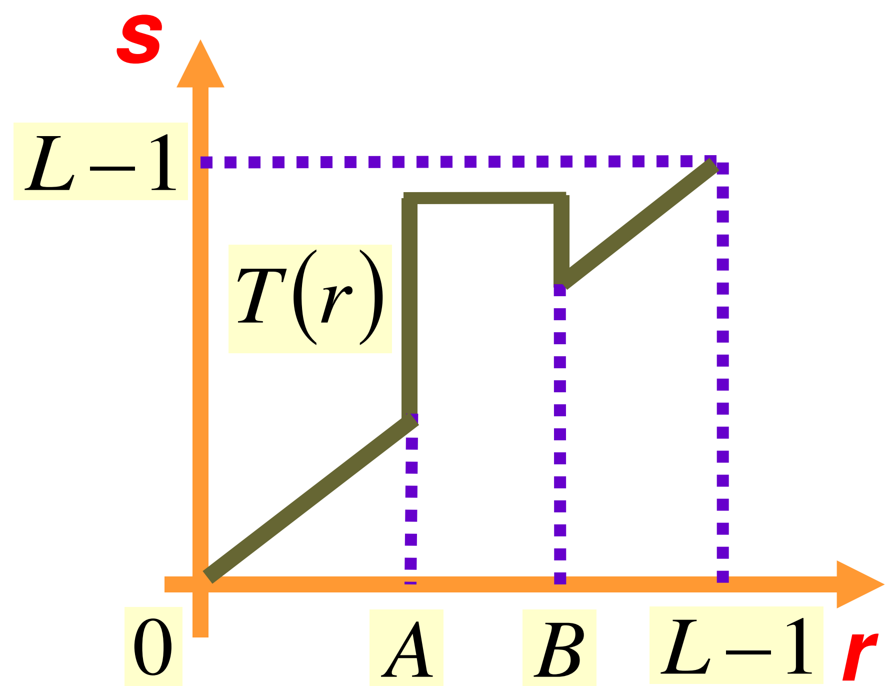
Spatial Domain Techniques

Point Processing Techniques

Gray-level Slicing - Approach 1



Gray-level Slicing - Approach 2



Spatial Domain Techniques

Point Processing Techniques

Comparison of Gray-level Slicing Approaches

An Aortic Angiogram.

Transformed image obtained with *slicing approach - 1*.

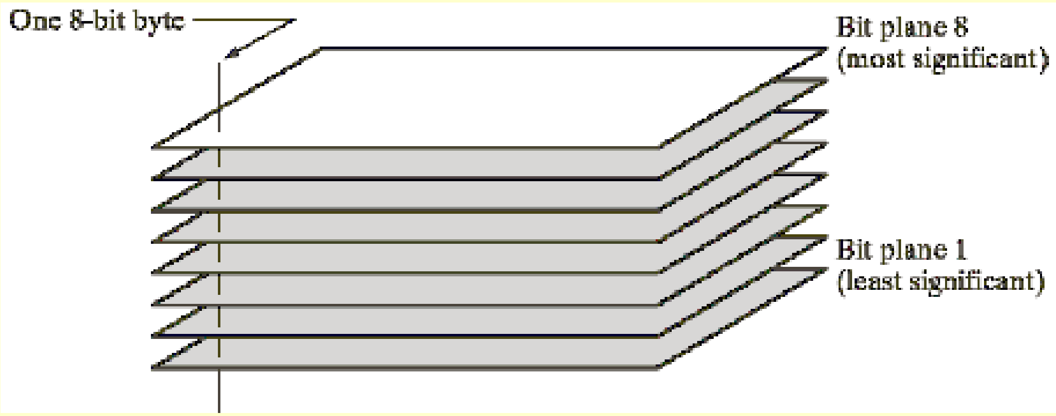
Transformed image obtained with *slicing approach - 2*.

Spatial Domain Techniques

Point Processing Techniques

Bit-Plane Slicing

- ✓ **Sometimes it is desirable to highlight the contribution made by specific bits to the total image appearance.**
- ✓ **The image can be imagined to be composed of eight 1-bit planes, plane 0 for the LSB to plane 7 for the MSB.**



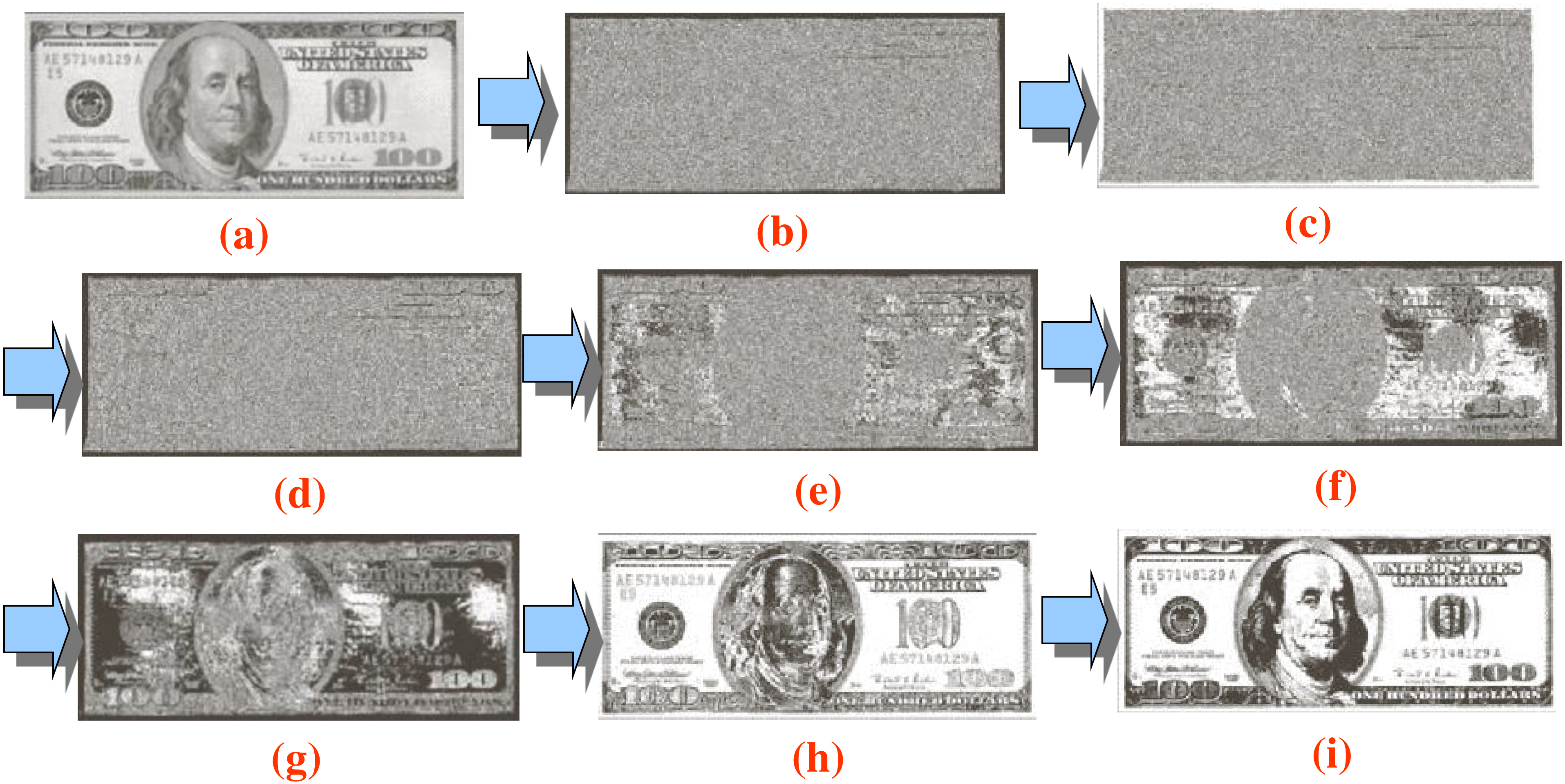
Bit-plane representation of an 8-bit digital image.

Highest order bits contain visually significant data.

Lowest order bits contain more subtle details.

Spatial Domain Techniques

Example of Bit-Plane Slicing



(a) An 8-bit gray scale image. (b) through (i) Bit planes 1 through 8 with bit plane 1 for LSB. Each bit plane is a binary image.

Spatial Domain Techniques

Point Processing Techniques

- ✓ In point-processing techniques, choice of different intensity transformation functions gives rise to different types of image enhancement.
- ✓ However, choice of a suitable function for a given input image is rather cumbersome and a trial-and-error based approach can be rather time consuming.

Solution ??

- ✓ An improved approach utilizes a systematic, automated solution for a suitable transformation mapping by employing information from image histogram. This can be achieved by utilizing:
    - o Histogram Equalization Technique, or
    - o Histogram Specification Technique.

Spatial Domain Techniques

Spatial Filtering

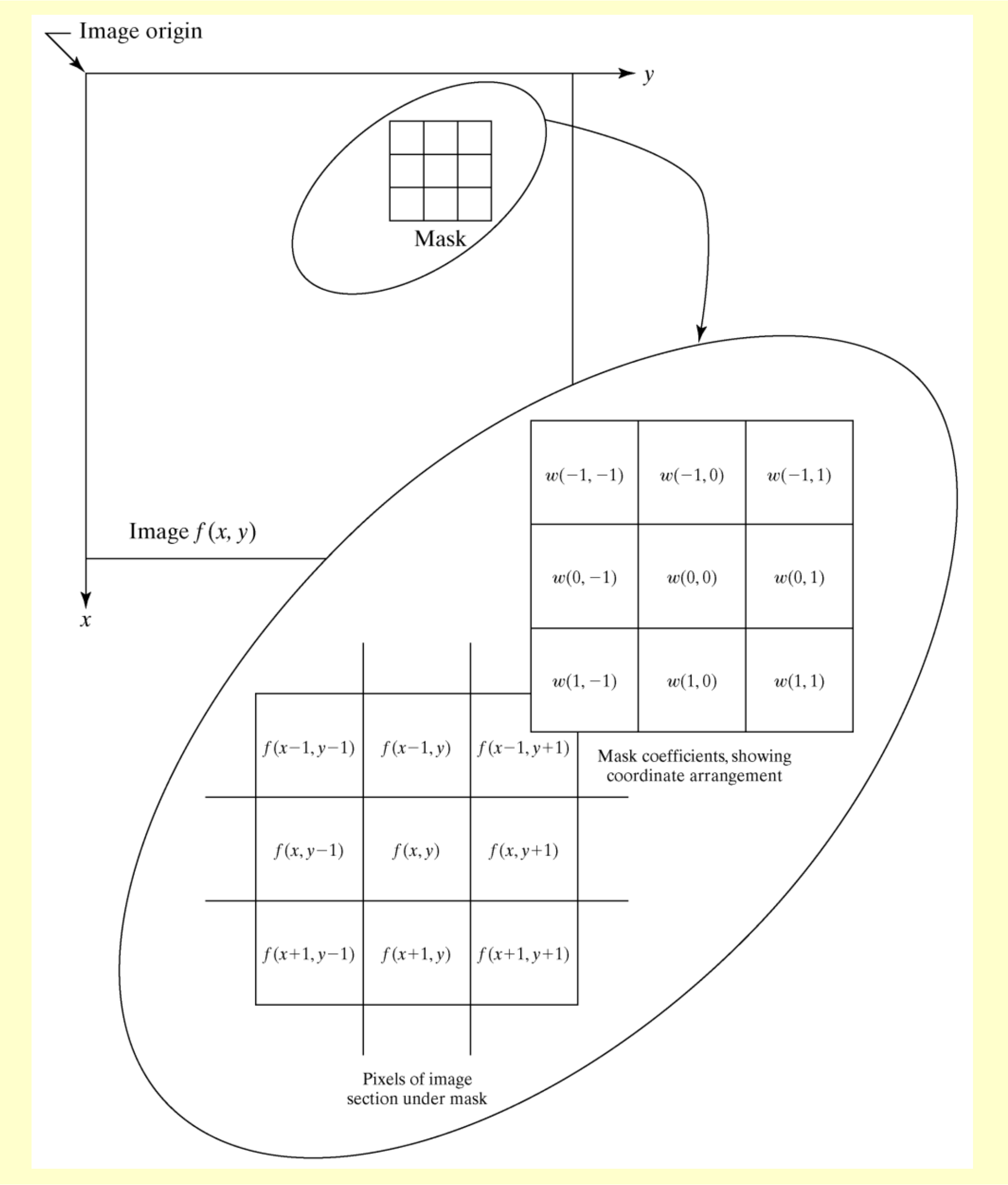
- ✓ ***Low pass filters*** **attenuate or eliminate high frequency components in the Fourier domain. Low pass filtering gives rise to image blurring.**
- ✓ ***High pass filters*** **attenuate or eliminate low frequency components in the Fourier domain. High pass filtering gives rise to sharpening of edges and other sharp details.**

How to Implement ??

- ✓ **Utilize suitable *two-dimensional masks* of suitable size e.g. 3 × 3, or 5 × 5, or 7 × 7, with appropriate mask co-efficients.**

Spatial Domain Techniques

Spatial Filtering



A 3 × 3 filter mask and the image neighborhood or sub-image under its influence.

Spatial Domain Techniques

Spatial Filtering

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

A 3 × 3 image section under consideration

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

A 3 × 3 filter mask employed for this image section

=

z_1	z_2	z_3
z_4	z_{5new}	z_6
z_7	z_8	z_9

Output of the filter for this image section

$$z_{5new} = f(R)$$

$$R = w_1 z_1 + w_2 z_2 + \Lambda + w_9 z_9 = \sum_{i=1}^{9} w_i z_i$$

***R* = Response of the mask**

The mask is centered on the image pixel whose new intensity value is to be calculated. This calculation is performed for each pixel separately by moving the mask to center it on the pixel under consideration.

Smoothing Spatial Filters

Low pass Spatial Filtering

$\frac{1}{9} \times$

1	1	1
1	1	1
1	1	1

(a)

$\frac{1}{16} \times$

1	2	1
2	4	2
1	2	1

(b)

Two typical 3 × 3 low pass filter masks

- ✓ **A *Low pass spatial filter* must have all positive coefficients.**
- ✓ **For a *Low pass spatial filter mask* shown above in (a), the operation is also popularly termed as neighborhood averaging. This averaging causes blurring and loss of sharpness.**
- ✓ **For a *filter mask* shown in (b), it is called weighted averaging.**

Smoothing Spatial Filters

Median Filtering

✓ ***Median filters*** **are nonlinear filters employed with an objective of noise reduction, without blurring.**

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

A 3 × 3 image section under consideration

Median filtering

z_1	z_2	z_3
z_4	z_{5new}	z_6
z_7	z_8	z_9

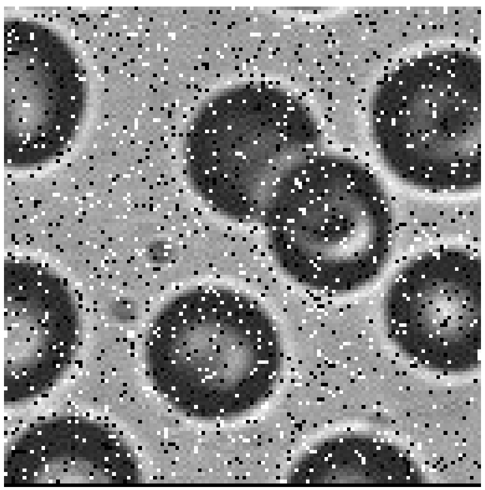
Output of the filter for this image section

$$z_{5new} = med\left(z_1, z_2, \Lambda\ , z_9\right)$$

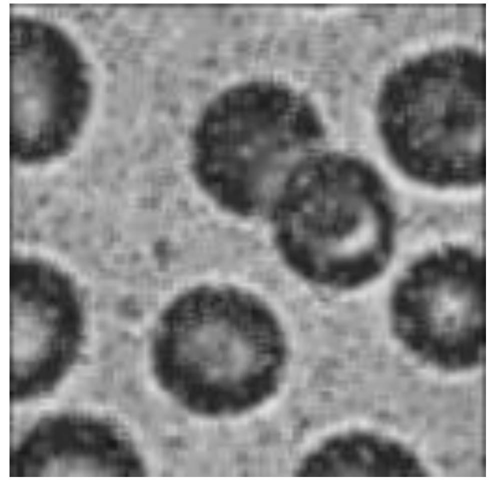
The filter is most effective when the noise pattern consists of spike-like components and it is of utmost importance to preserve edge sharpness.

Smoothing Spatial Filters

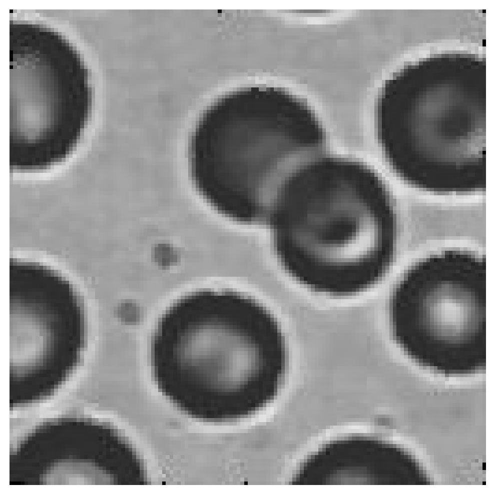
A Comparison of Average and Median Filtering - I



A Blood cell image corrupted by salt-and-pepper noise.



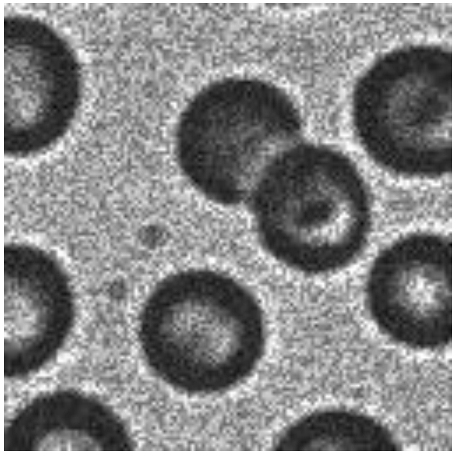
Processed image employing a 3 × 3 averaging mask.



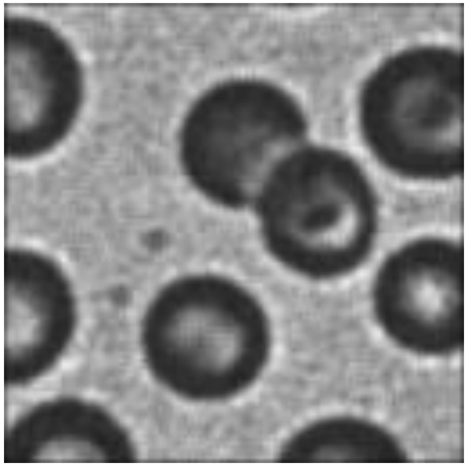
Processed image employing a 3 × 3 median filter.

Smoothing Spatial Filters

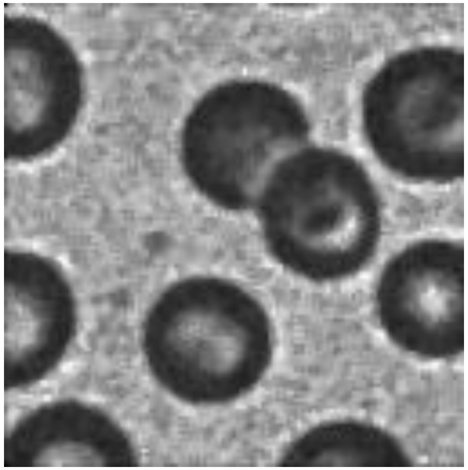
A Comparison of Average and Median Filtering - II



Same Blood cell image corrupted by speckle noise.



Processed image employing a 3 × 3 averaging mask.



Processed image employing a 3 × 3 median filter.

Sharpening Spatial Filters

Derivative Filters

- ✓ **The *differentiation operation* is expected to sharpen an image.**
- ✓ **One can use either *first derivative* or *second derivative* information.**

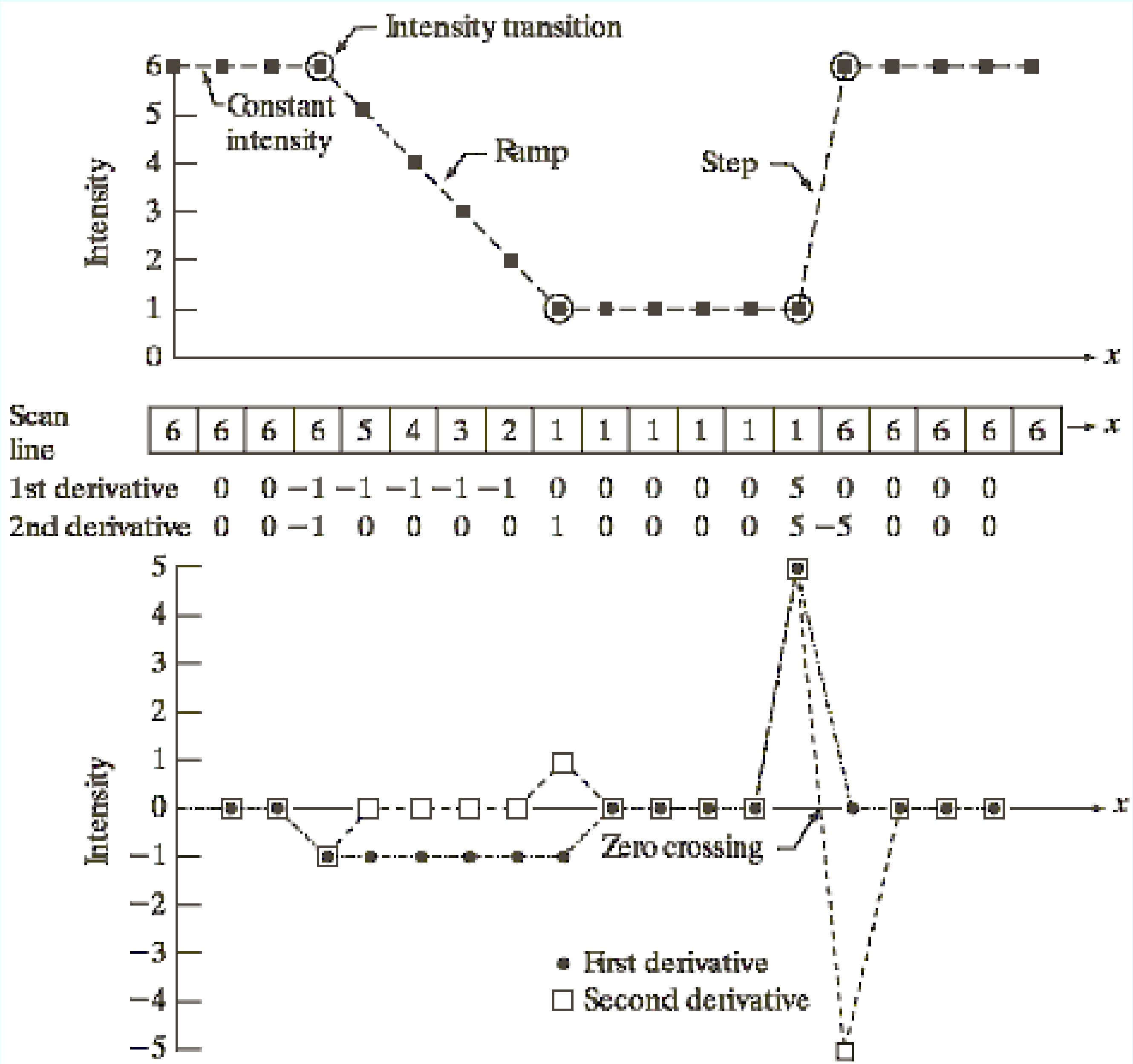
Digital Approximation of First Derivative:

$$\frac{\partial f(x,y)}{\partial x} = f(x+1,y) - f(x,y)$$

Digital Approximation of Second Derivative:

$$\frac{\partial^2 f(x,y)}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y)$$

Sharpening Spatial Filters



First and second derivatives of a horizontal, single-row intensity profile of an image.

Sharpening Spatial Filters

Constraints of using a Definition for First Derivative Filters

- ✓ **Must be zero in areas of** ***constant intensity.***
- ✓ **Must be non-zero at the** ***onset of an intensity step or ramp.***
- ✓ **Must be non-zero** ***along ramps.***

Constraints of using a Definition for Second Derivative Filters

- ✓ **Must be zero in areas of** ***constant intensity.***
- ✓ **Must be non-zero at the** ***onset and end of an intensity step or ramp.***
- ✓ **Must be zero** ***along ramps of constant slope.***

Sharpening Spatial Filters

Implementing a First Derivative Filter for Image Sharpening

By applying the *Gradient*.

The gradient of a function $f(x, y)$ at coordinates (x, y) is defined as the two-dimensional column vector:

$$\nabla f \equiv grad(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

The magnitude (length) of vector ∇f:

$$M(x, y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2}$$

$$M(x, y) \approx |g_x| + |g_y|$$

$M(x, y)$ is an image of same size as the original and called the gradient image. The computation of this gradient is the basis for various approaches to develop first derivative filter.

Sharpening Spatial Filters

First Derivative Filters

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

A 3 × 3 image section under consideration

Derivative filtering

$$M(x,y)=\left[(z_8-z_5)^2+(z_6-z_5)^2\right]^{1/2}$$

Approximated form:

$$M(x,y)\approx|z_8-z_5|+|z_6-z_5|$$

Another approach using cross-differences:

$$M(x,y)\approx\left[(z_9-z_5)^2+(z_8-z_6)^2\right]^{1/2}$$

$$\nabla f\approx|z_9-z_5|+|z_8-z_6|$$

These equations can be implemented using masks of size 2 × 2 .

✓ ***Constraint:*** **Masks of even sizes are awkward to implement. Hence an approximation with 3 × 3 neighborhood is preferred.**

Sharpening Spatial Filters

First Derivative Filters

The masks for 1st derivative operators of size 3 × 3:

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Sobel operators

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

$$M(x, y) \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|$$

Sharpening Spatial Filters

First Derivative Filters

The masks for 1st derivative operators of size 3 × 3:

-1	-1	-1
0	0	0
1	1	1

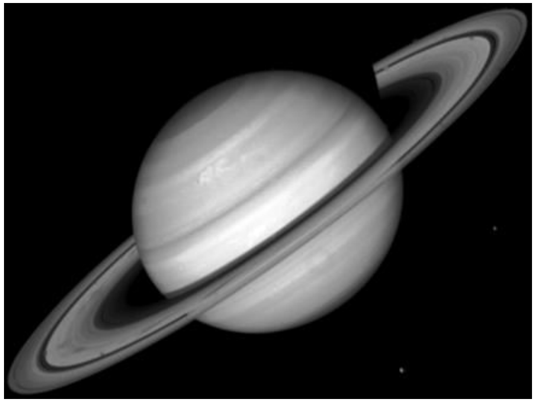
-1	0	1
-1	0	1
-1	0	1

Prewitt operators

$$M(x,y) \approx \left|(z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)\right| + \left|(z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)\right|$$

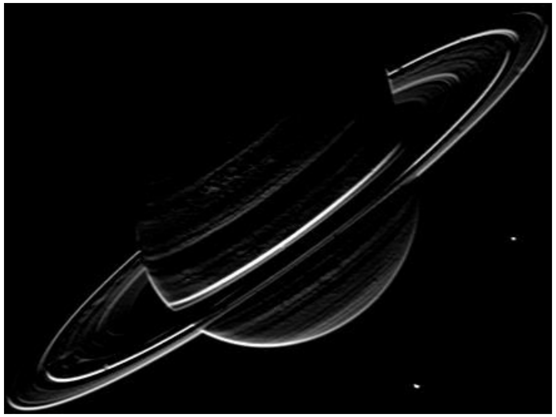
Sharpening Spatial Filters

Example of Sobel Masks



Original 'Saturn' image.

Blurred 'Saturn' image.



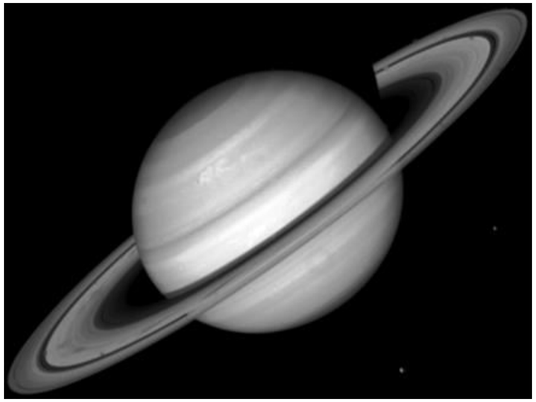
Output of *Sobel Vertical Mask*.

Output of *Sobel Horizontal Mask*.

Output of *Sobel Gradient Mask*.

Sharpening Spatial Filters

Example of Sobel Masks – contd …



Original 'Saturn' image.

Blurred 'Saturn' image.

Output of *Sobel Gradient Mask*.

Output of *Sobel Gradient Mask added to the Blurred image*.

Sharpening Spatial Filters

Implementing a Second Derivative Filter for Image Sharpening

By applying the *Laplacian*.

The Laplacian of an image function *f*(*x*, *y*) of two variables:

$$\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

Common digital approximations of the second derivatives:

$$\frac{\partial^2 f(x,y)}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y)$$

$$\frac{\partial^2 f(x,y)}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y)$$

$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

Sharpening Spatial Filters

Second Derivative Filters

The masks for 2nd derivative operators of size 3 × 3:

0	1	0
1	-4	1
0	1	0

Mask 1

Mask employing Laplacian considering conventional horizontal and vertical directions

1	1	1
1	-8	1
1	1	1

Mask 2

Mask employing Laplacian considering four directions: (a) horizontal, (b) vertical, (c) +45° and (c) -45° directions

Sharpening Spatial Filters

Second Derivative Filters

The masks for 2nd derivative operators of size 3 × 3:

0	-1	0
-1	4	-1
0	-1	0

Mask 3

A Laplacian Mask employing negative of the Laplacian Mask 1

-1	-1	-1
-1	8	-1
-1	-1	-1

Mask 4

A Laplacian Mask employing negative of the Laplacian Mask 2

Sharpening Spatial Filters

Second Derivative Filters

There is a Problem …

The Laplacian is a derivative operator that highlights intensity discontinuities in an image and, in the process, de-emphasizes image regions having slow variations in intensity profile.

How to preserve the original background features and yet perform sharpening operation ??

Utilize the Laplacian in the following manner:

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

$f(x, y)$: The input image

$g(x, y)$: The sharpened output image

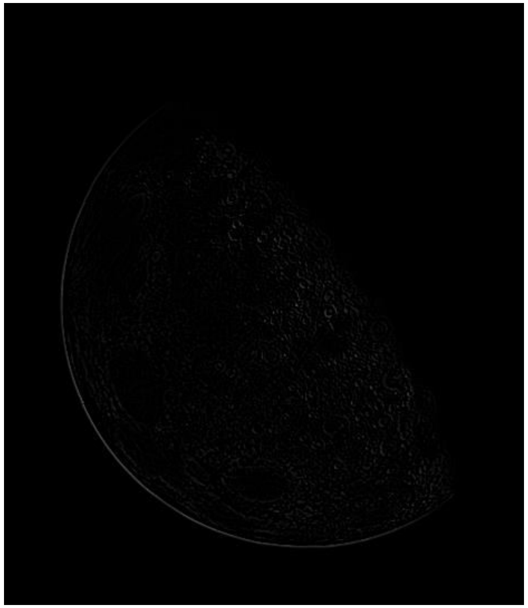
c: a constant

c = -1, for Masks 1 and 2. c = +1, for Masks 3 and 4.

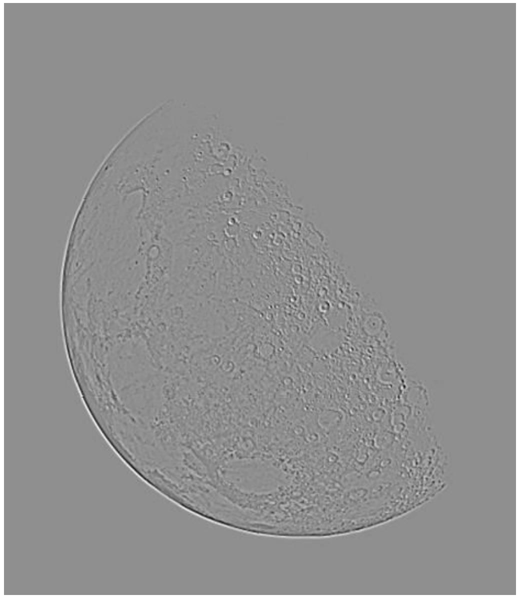
Sharpening Spatial Filters

Example of Laplacian Filters

'Blurry Moon' image.



Output of *Laplacian Mask 1 (without scaling)*.



Output of *Laplacian Mask 1 (with scaling)*.

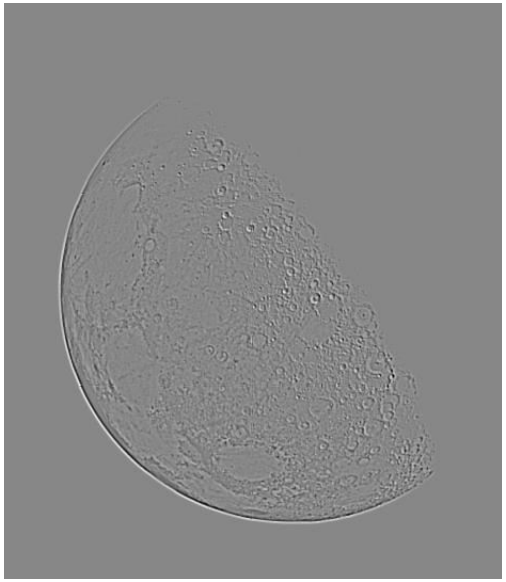
Output of *non-scaled Mask 1 subtracted from the Blurry image*.

Sharpening Spatial Filters

Example of Laplacian Filters – contd…

'Blurry Moon' image.

Output of *Laplacian Mask 2 (without scaling)*.



Output of *Laplacian Mask 2 (with scaling)*.

Output of *non-scaled Mask 2 subtracted from the Blurry image*.

Sharpening Spatial Filters

Performance Comparison of Laplacian Filters

'Blurry Moon' image.

Performance of the *Laplacian Mask 1*.

Performance of the *Laplacian Mask 2*.

Sharpening Spatial Filters

High pass Spatial Filtering using First and Second Derivatives

An Important point …

- ✓ **For a *high pass spatial filter mask,* whether utilizing first derivative or second derivative, the sum of the mask coefficients is always zero.**

Sharpening Spatial Filters

Unsharp Masking and High –boost Filtering

Create a blurred version of the original image → $f(x,y) \Rightarrow \bar{f}(x,y)$

↓

Subtract the blurred version from the original → $g_{mask}(x,y) = f(x,y) - \bar{f}(x,y)$

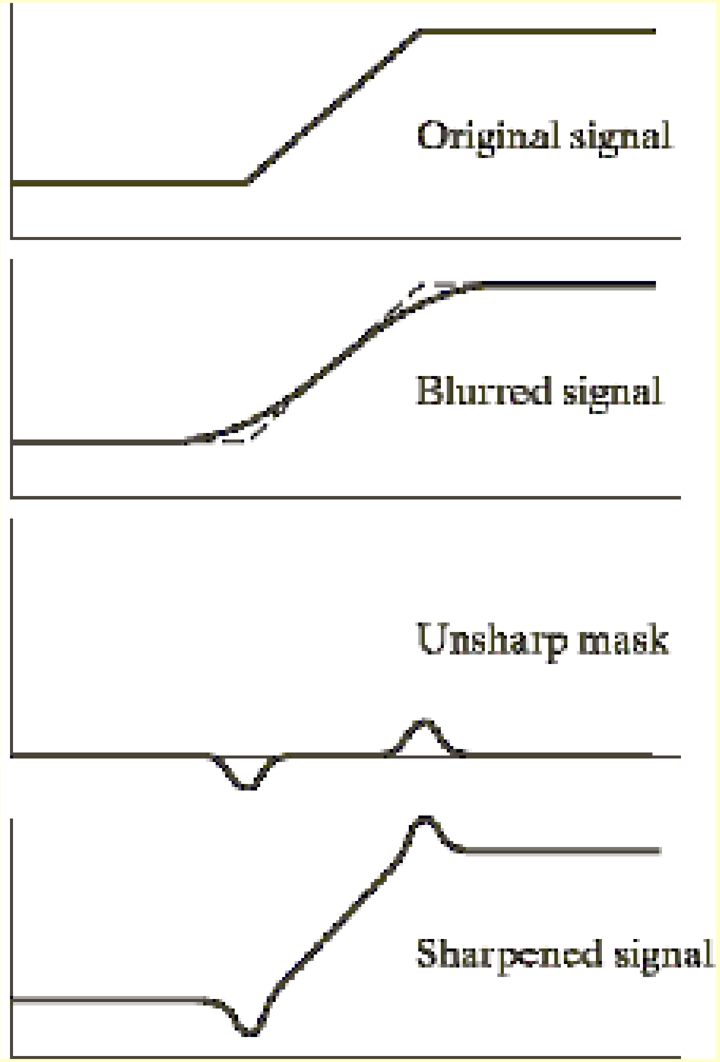
↓

Add this mask to the original image → $g(x,y) = f(x,y) + k * g_{mask}(x,y)$

- ✓ **When $k = 1$, we have *unsharp masking*.**
- ✓ **When $k > 1$, we have *high-boost filtering*.**

Sharpening Spatial Filters

Illustration of Unsharp Masking



Frequency Domain Methods

- ✓ **Frequency domain techniques for image enhancement utilize *convolution theorem in two-dimensions*.**
- ✓ **The *two-dimensional Discrete Fourier Transform (DFT) pair* is given as:**

$$F(u,v)=\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\,\mathbf{exp}\left[-j2\pi(ux/M+vy/N)\right]$$

$$u=0,1,2,\Lambda\,,M-1 \quad \text{and} \quad v=0,1,2,\Lambda\,,N-1$$

and

$$f(x,y)=\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}F(u,v)\,\mathbf{exp}\left[j2\pi(ux/M+vy/N)\right]$$

$$x=0,1,2,\Lambda\,,M-1 \quad \text{and} \quad y=0,1,2,\Lambda\,,N-1$$

Frequency Domain Methods

Convolution Theorem in Two-Dimensions:

$$f(x,y)*g(x,y) \Leftrightarrow F(u,v)G(u,v)$$

and

$$f(x,y)g(x,y) \Leftrightarrow F(u,v)*G(u,v)$$

In Image Enhancement Problems:

$$g(x,y) = h(x,y)*f(x,y)$$

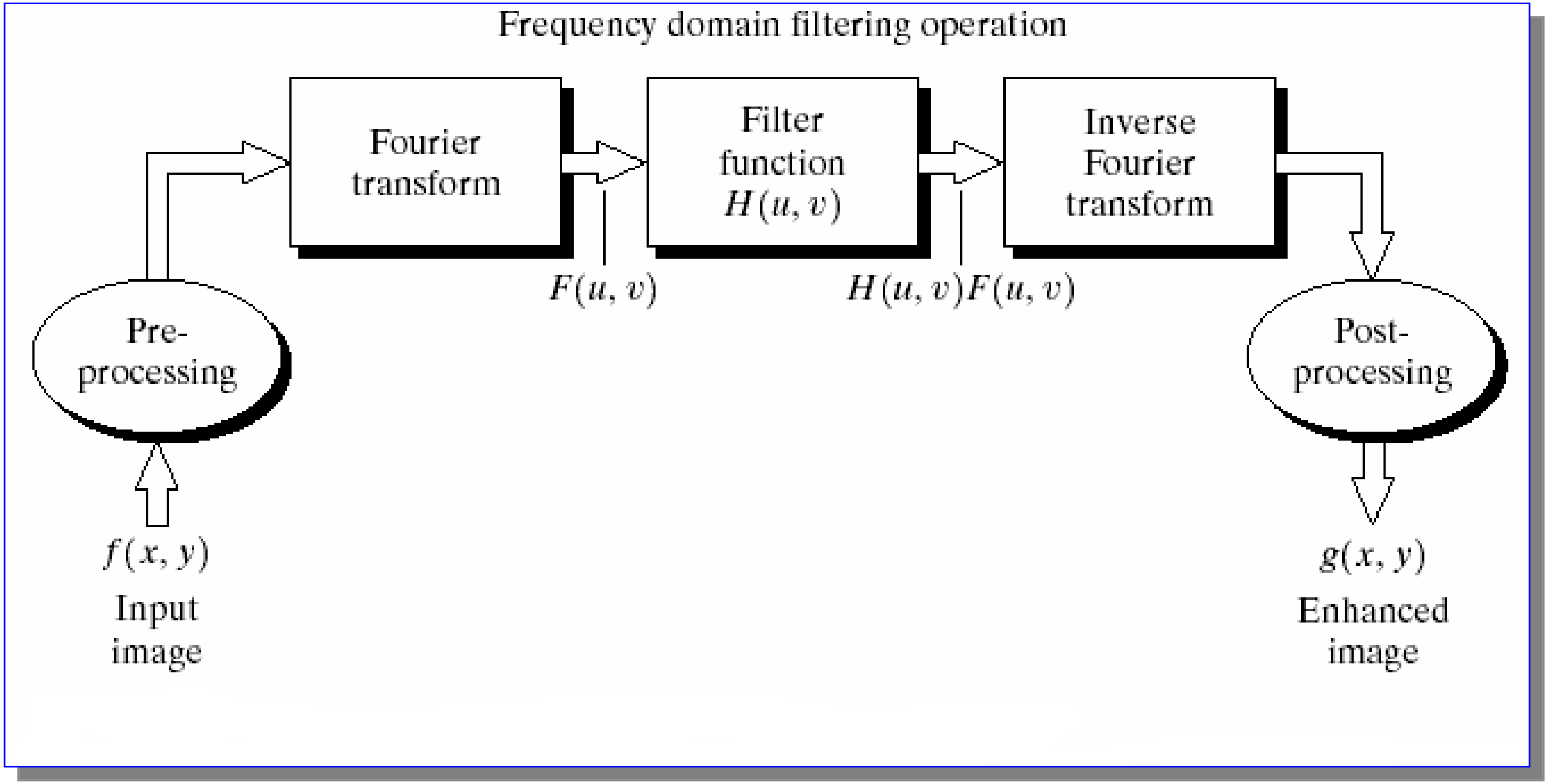
$$G(u,v) = H(u,v)F(u,v)$$

$h(x, y)$: input image
$f(x, y)$: a linear, position invariant operator
$g(x, y)$: output image

G, H and F are FTs of g, h, and f. $H(u, v)$ is the process transfer function.

Our Problem:
$f(x, y)$ is given. Select $H(u, v)$ such that $g(x, y)$ exhibit some highlighted feature of $f(x, y)$.

Frequency Domain Filtering



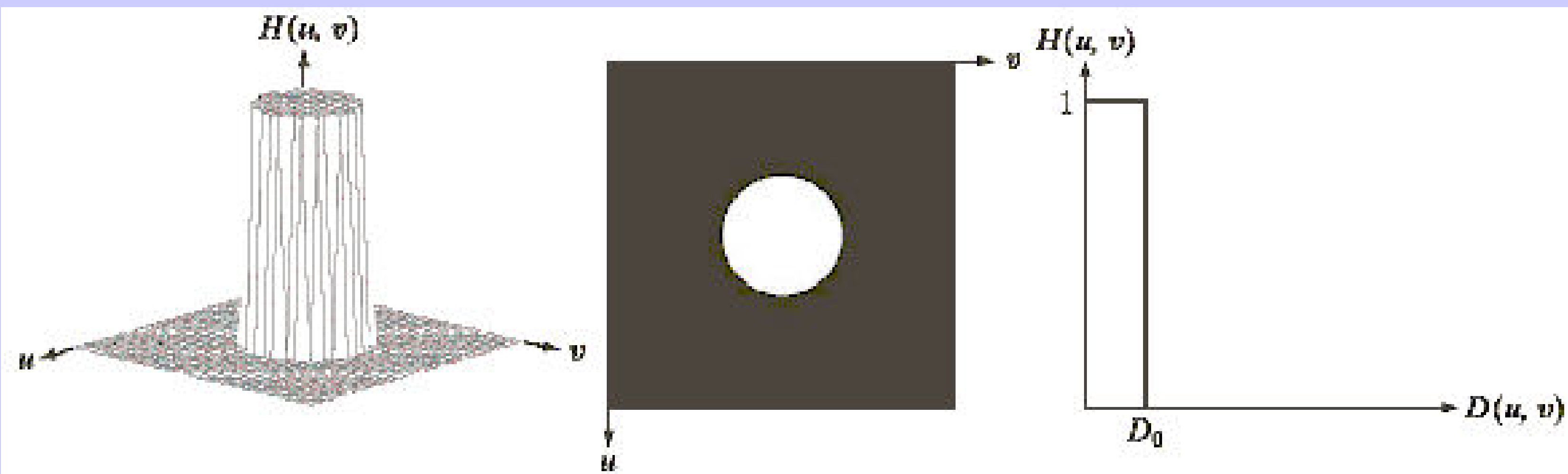
Ideal Lowpass Filter (ILPF)

Transfer function of a 2-D ILPF:

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

D_0: a specified non-negative quantity;
$D(u, v)$: Distance of the point (u, v) from the origin of the frequency plane

$$D(u,v)=\sqrt{u^2+v^2}$$



(a) A 3-D perspective plot of an ILPF T.F., (b) Filter displayed as an image, (c) Filter radial cross section.

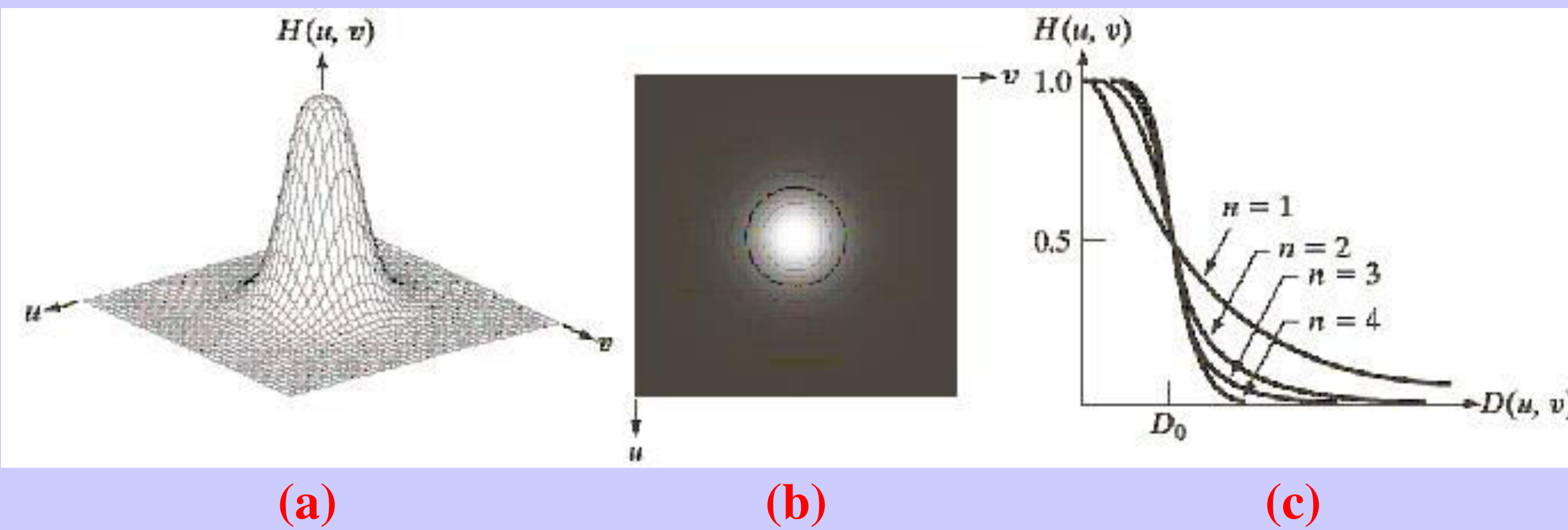
Butterworth Lowpass Filter (BLPF)

Transfer function of a 2-D BLPF:

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

n is the order of the filter. The cut-off frequency locus is at a distance D_0 from the origin.

$$D(u,v) = \sqrt{u^2 + v^2}$$



(a) A 3-D perspective plot of a BLPF T.F., (b) Filter displayed as an image, (c) Filter radial cross section of orders 1-4.

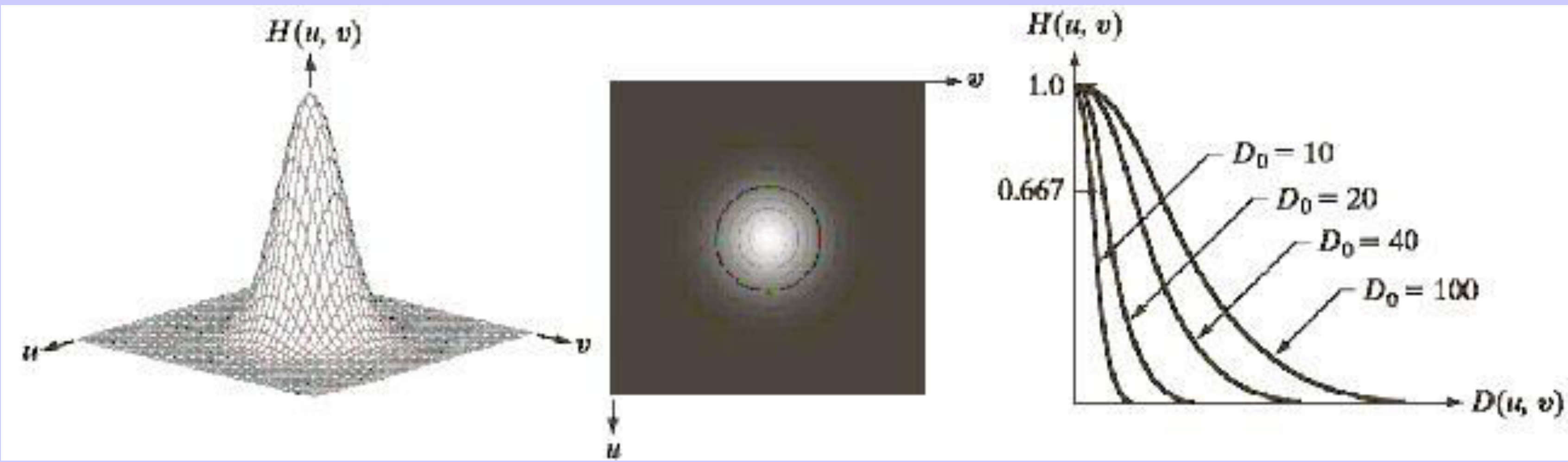
Gaussian Lowpass Filter (GLPF)

Transfer function of a 2-D GLPF:

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

$$D(u,v) = \sqrt{u^2 + v^2}$$

σ is a measure of the spread of the Gaussian curve.
The cut-off frequency locus is at a distance D_0 from the origin.



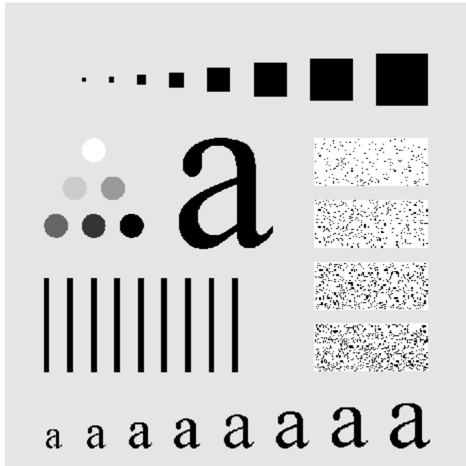
(a) (b) (c)

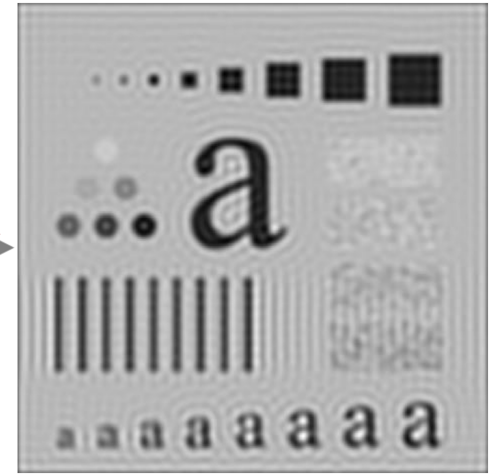
(a) A 3-D perspective plot of a GLPF T.F., (b) Filter displayed as an image, (c) Filter radial cross section for various D_0.

Lowpass Filters

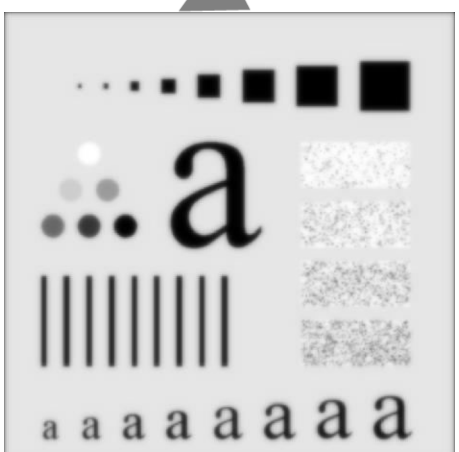
Comparison of Ideal, Butterworth and Gaussian Filtering - I

Original image of a test pattern.





ILPF with D_0 = 80



BLPF with n = 1, D_0 = 80

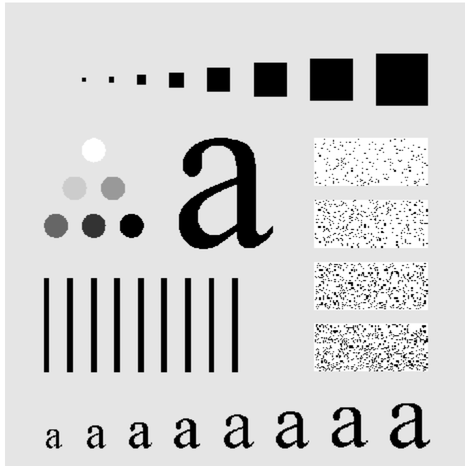
BLPF with n = 2, D_0 = 80

GLPF with D_0 = 80

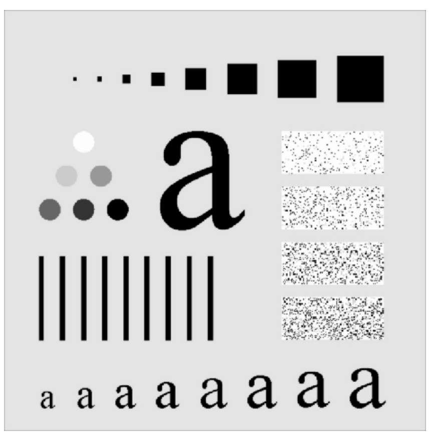
Lowpass Filters

Comparison of Ideal, Butterworth and Gaussian Filtering - II

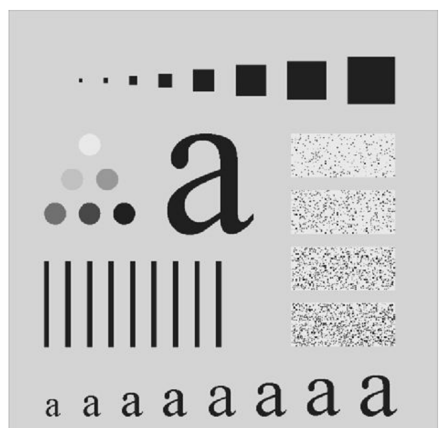
Original image of a test pattern.



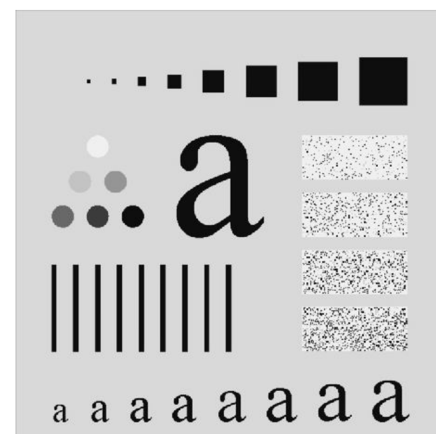
ILPF with D_0 = 480



BLPF with n = 1, D_0 = 480



BLPF with n = 2, D_0 =480



GLPF with D_0 = 480

Ideal Highpass Filter (IHPF)

Transfer function of a 2-D IHPF:

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

$$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

D_0: specified cut-off distance from the origin of the frequency plane;
$D(u, v)$: Distance of the point (u, v) from the origin of the frequency plane

Butterworth Highpass Filter (BHPF)

Transfer function of a 2-D BHPF:

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

$$D(u,v) = \sqrt{u^2 + v^2}$$

n is the order of the filter. The cut-off frequency locus is at a distance D_0 from the origin.

Gaussian Highpass Filter (GHPF)

Transfer function of a 2-D GHPF:

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

$$D(u,v) = \sqrt{u^2 + v^2}$$

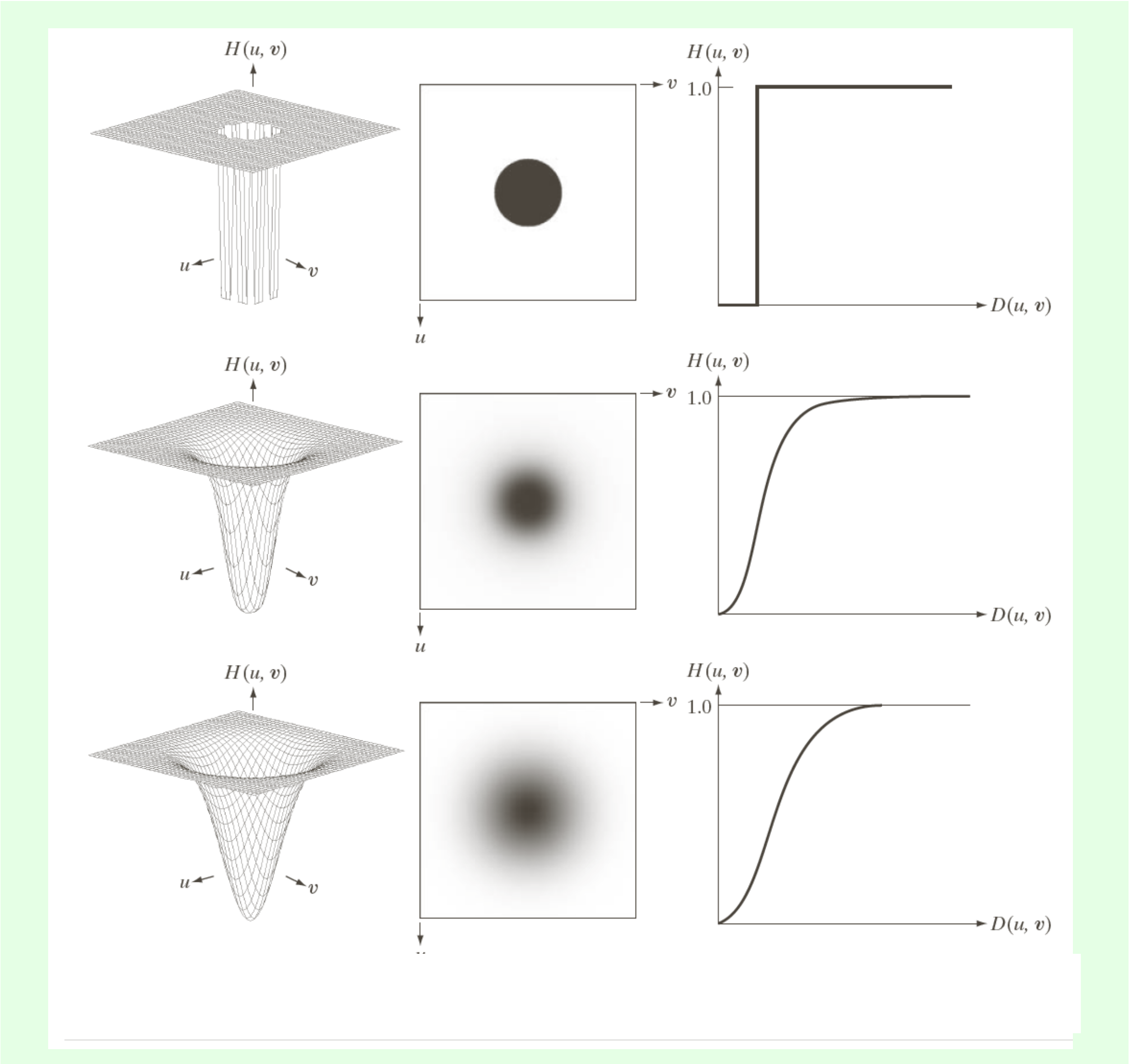
The cut-off frequency locus is at a distance D_0 from the origin.

Highpass Filters

(a)

(b)

(c)

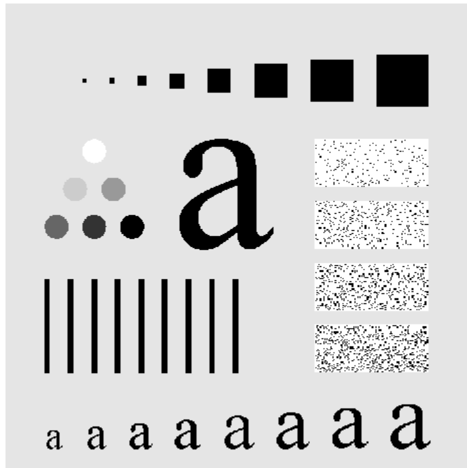


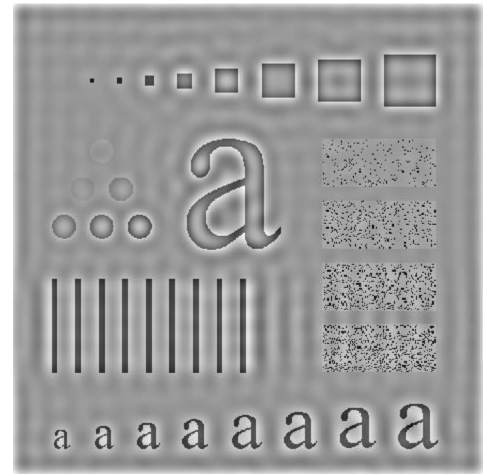
3-D perspective plot, image representation, and radial cross section for (a) IHPF, (b) BHPF, and (c) GHPF.

Highpass or Sharpening Filters

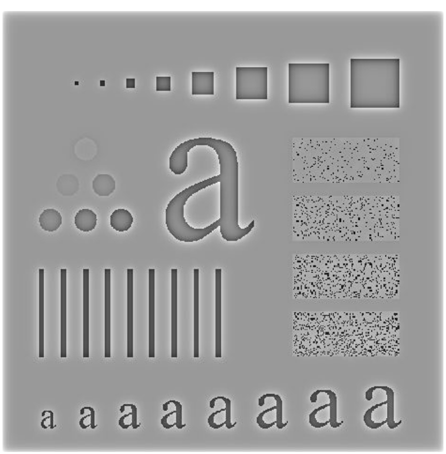
Comparison of Ideal, Butterworth and Gaussian Filtering - I

Original image of a test pattern.

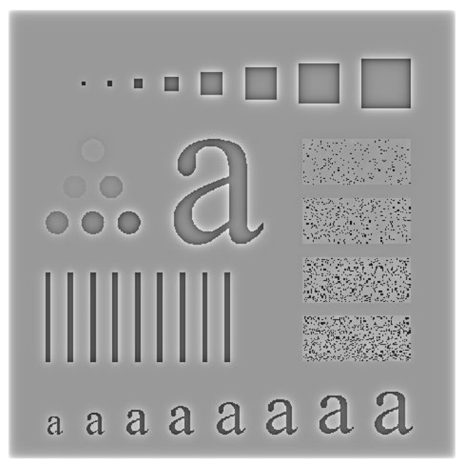




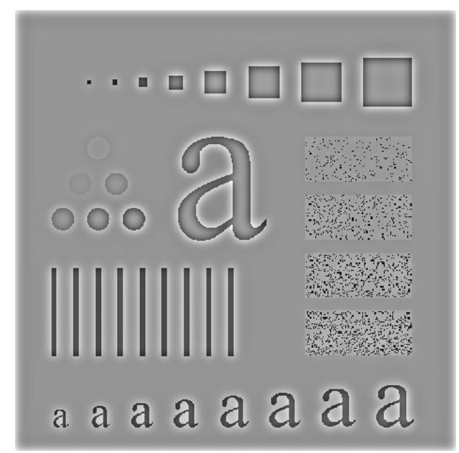
IHPF with D_0 = 40



BHPF with n = 1, D_0 = 40



BHPF with n = 2, D_0 = 40

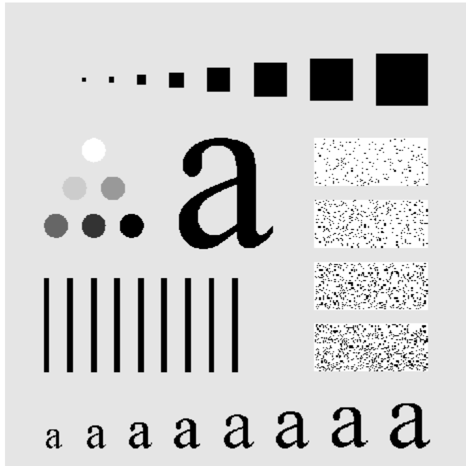


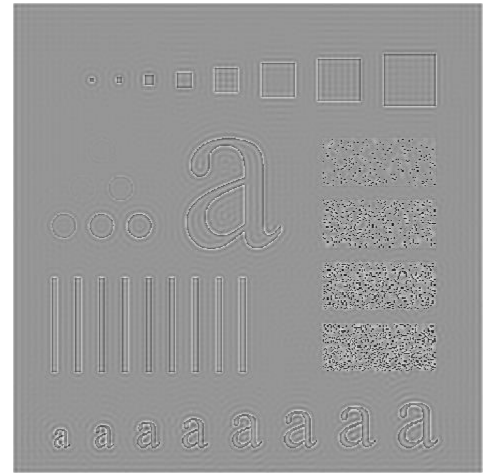
GHPF with D_0 = 40

Highpass or Sharpening Filters

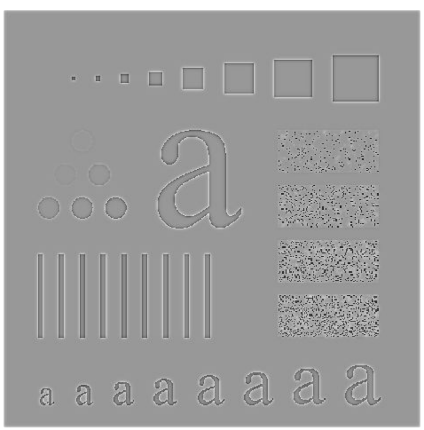
Comparison of Ideal, Butterworth and Gaussian Filtering - II

Original image of a test pattern.

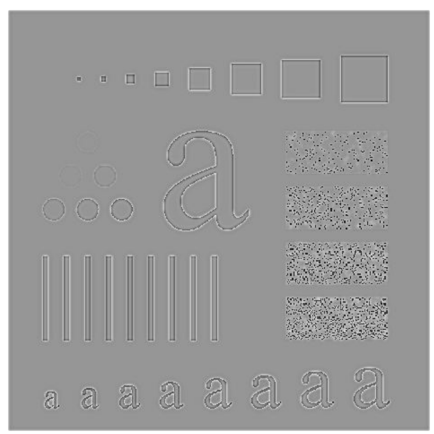




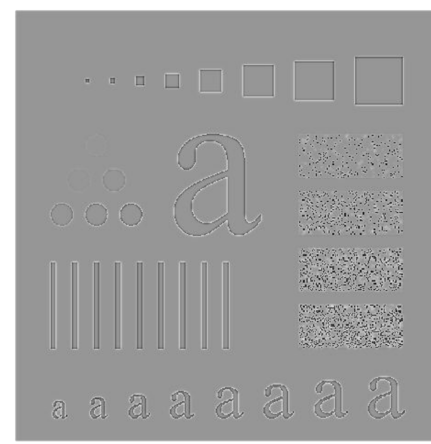
IHPF with D_0 = 150



BHPF with n = 1, D_0 = 150



BHPF with n = 2, D_0 = 150



GHPF with D_0 = 150

High-Frequency Emphasis Filtering (HFEF)

Transfer function of a HFEF:

$$H_{hfe}(u,v) = a + b * H_{hp}(u,v)$$

a: offset; ***b***: multiplier;

$H_{hp}(u, v)$: High pass filter T.F.

The offset term is used to preserve the low frequency components.

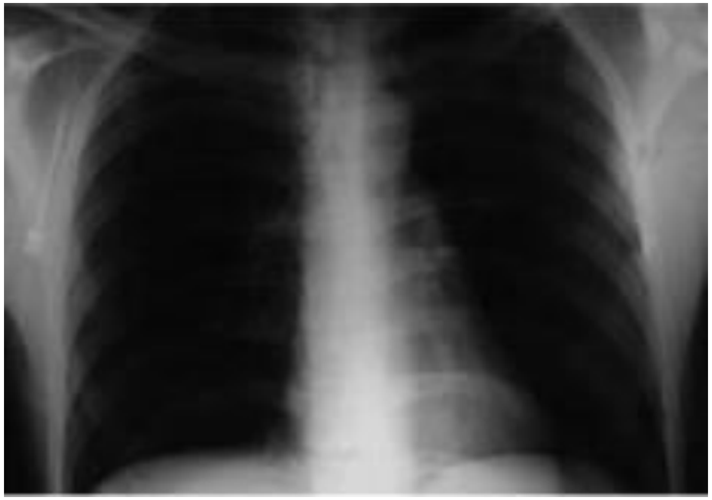
The multiplier term is used to highlight the high frequency components.

- ***Note:*** This technique requires a post-filtering processing to redistribute the gray levels.

Candidate tool for post-filtering ??

- **Histogram equalization** is a popular option because it can provide contrast enhancement.

High-Frequency Emphasis Filtering (HFEF)



A chest X-ray image.

Processed image employing BHPF (n = 2, D_0 = 5% of image vertical dimension).

Processed image after high-frequency emphasis filtering (a = 0.5, b = 2.0).

Processed image after HFEF, followed by histogram equalization.

Homomorphic Filtering

✓ **This method utilizes the illumination-reflectance model.**

$$f(x,y)=i(x,y)r(x,y)$$

$i(x, y)$: illumination component;
$r(x, y)$: reflectance component.

There is a problem …

✓ **This equation can not be used directly to operate separately on the frequency components of illumination and reflectance.**

Why ??

✓ **Because the Fourier Transform of the product of the two functions is not separable.**

$$\mathcal{F}\{f(x,y)\}\neq\mathcal{F}\{i(x,y)\}\mathcal{F}\{r(x,y)\}$$

Homomorphic Filtering (contd…)

Let us define:

$$z(x,y)=\ln f(x,y)=\ln i(x,y)+\ln r(x,y)$$

$$\mathcal{F}\{z(x,y)\}=\mathcal{F}\{\ln f(x,y)\}=\mathcal{F}\{\ln i(x,y)+\ln r(x,y)\}$$

$$Z(u,v)=I(u,v)+R(u,v)$$

***I*(*u*, *v*): F.T. of *ln i*(*x*, *y*);**

***R*(*u*, *v*): F.T. of *ln r*(*x*, *y*).**

Processing *Z*(*u*, *v*) with a Homomorphic Filter T.F. *H*(*u*, *v*):

$$S(u,v)=H(u,v)Z(u,v)=H(u,v)I(u,v)+H(u,v)R(u,v)$$

$$s(x,y)=\mathcal{F}^{-1}\{S(u,v)\}=\mathcal{F}^{-1}\{H(u,v)I(u,v)\}+\mathcal{F}^{-1}\{H(u,v)R(u,v)\}$$
$$=i'(x,y)+r'(x,y)$$

Homomorphic Filtering (contd…)

- As $z(x, y)$ was formed by taking the logarithm of the original image $f(x, y)$, the inverse operation on $s(x, y)$ should give us the desired enhanced image $g(x, y)$.

$$
\begin{aligned}
g(x, y) &= \exp[s(x, y)] \\
&= \exp[i'(x, y)]\exp[r'(x, y)] \\
&= i_0(x, y) r_0(x, y)
\end{aligned}
$$

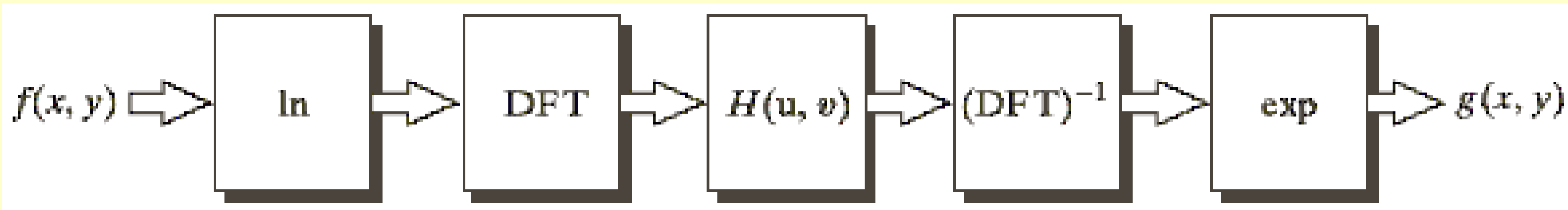
$i_0(x, y)$: illumination component of the output image;

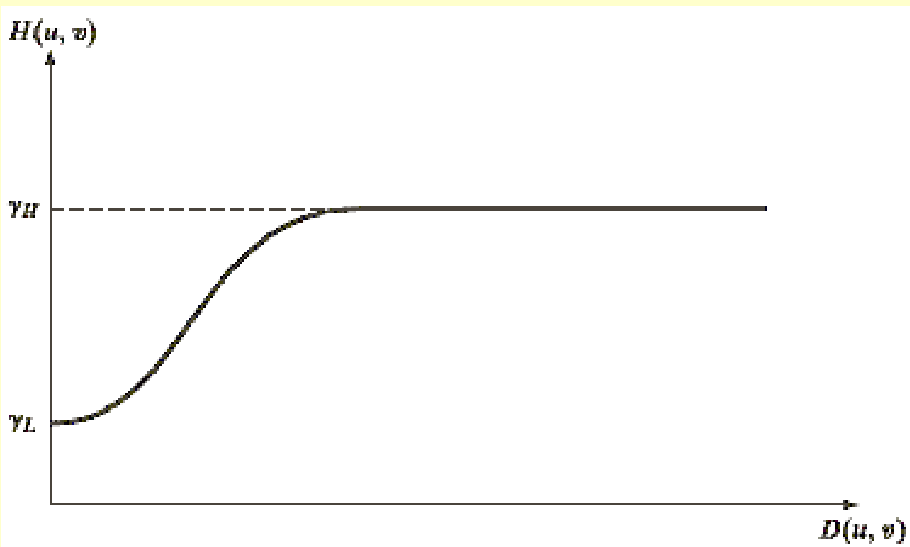
$r_0(x, y)$: reflectance component of the output image;

What is the salient feature of this approach ??

This approach works on separating the illumination and reflectance components of an image so that the Homomorphic Filter T.F. can work on these components separately.

Homomorphic Filtering (contd…)



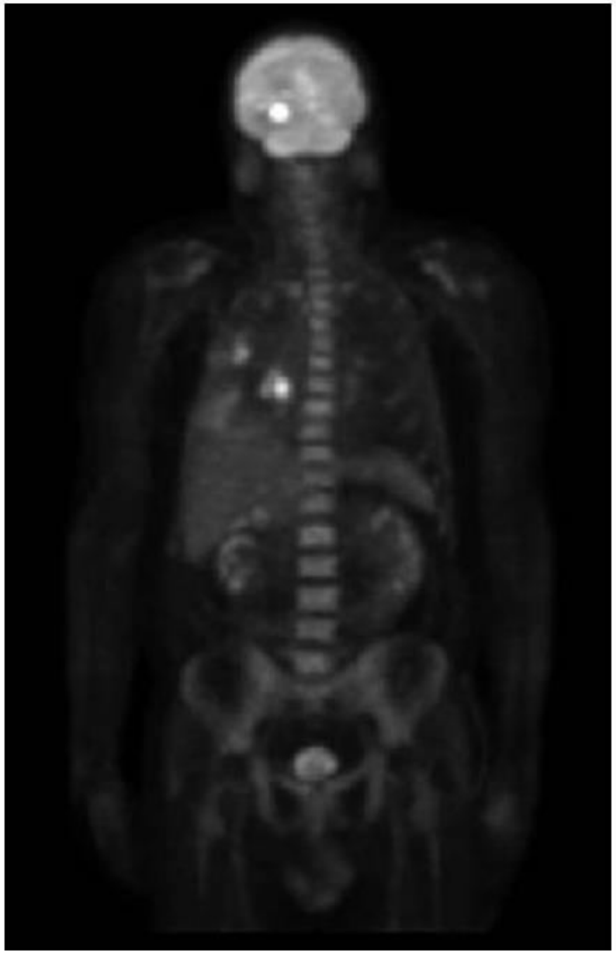


Illumination Component is characterized by slow spatial variations.

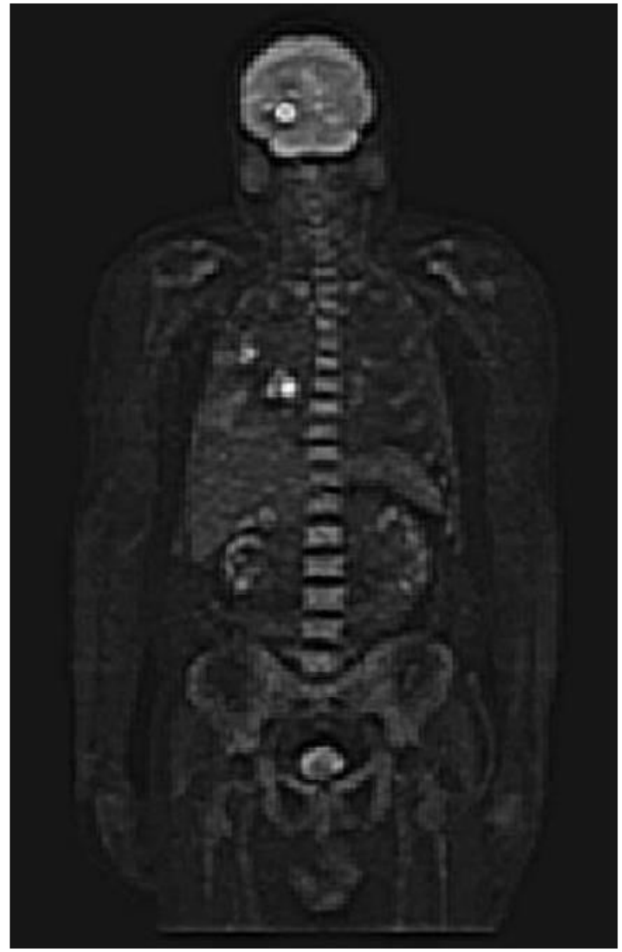
Reflectance Component is characterized by abrupt variations.

Cross section of a circularly symmetric Homomorphic Filter function.

Homomorphic Filtering (contd…)



A full body PET (Positron Emission Tomography) scan.



Homomorphic Filtered enhanced image.

Image Enhancement Techniques

References:

- R. C. Gonzalez and R. E. Woods. Digital Image Processing. Pearson Education Inc., 2008.
- R. C. Gonzalez, R. E. Woods, and S. L. Eddins. Digital Image Processing using MATLAB®. Pearson Education, Inc. 2005.
- S. Annadurai and R. Shanmugalakshmi. Fundamentals of Digital Image Processing. Pearson Education, Inc. 2007.

Thank You…