Final Exam Review

Image enhancement

- Image is NOT Perfect in many situations
- Two general categories of problems
  - An image needs improvement
  - Low-level features must be detected

- **Definition: *Image enhancement*** operators improve the detectability of important image details or objects.
  - Example operations include noise reduction, smoothing, contrast stretching, and edge enhancement.

Input image → **Enhancement technique** → Better image

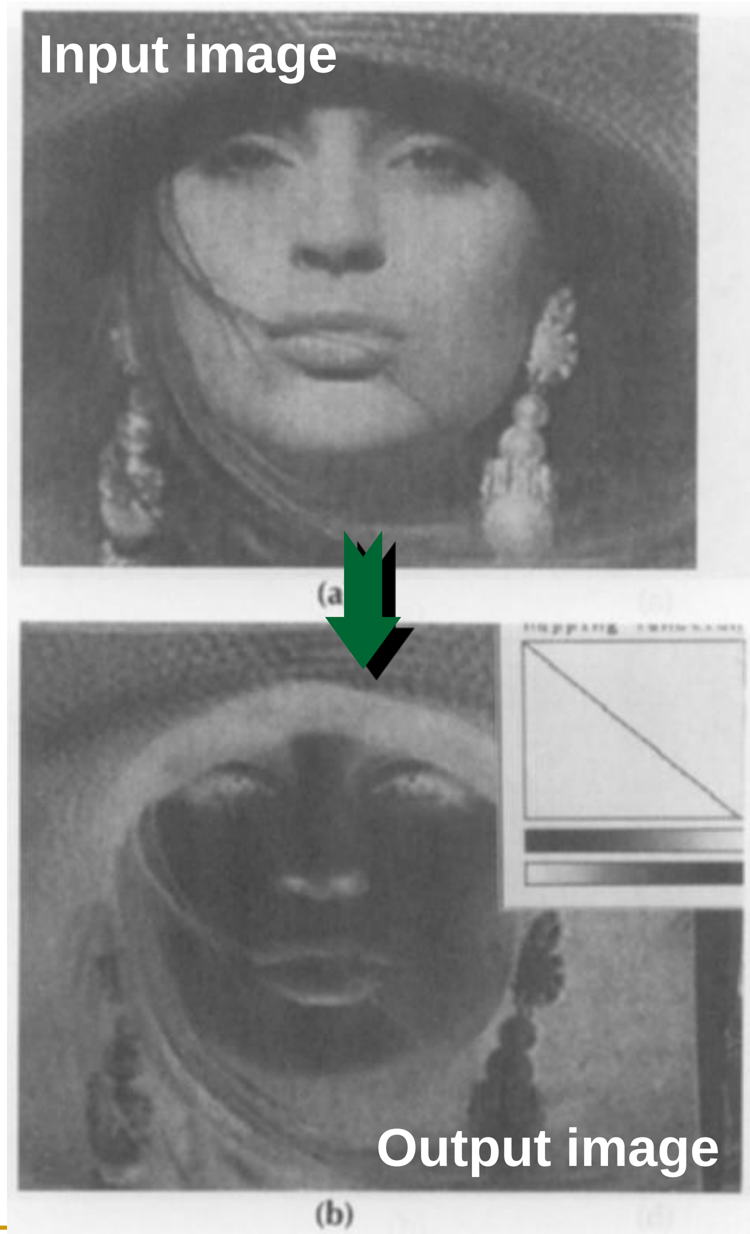
No general theory

Application specific

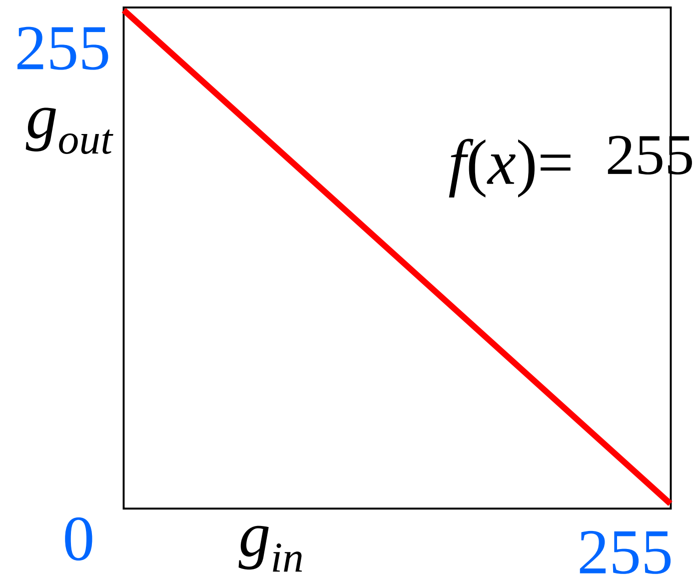
Spatial domain
Manipulate pixel intensity directly

Frequency domain
Modify the Fourier transform

- A **contrast stretching** **(對比度伸縮)** operator is a point operator that uses a piecewise smooth function $f\,(\mathrm{In}[x, y])$ of the input gray level to enhance important details of the image.

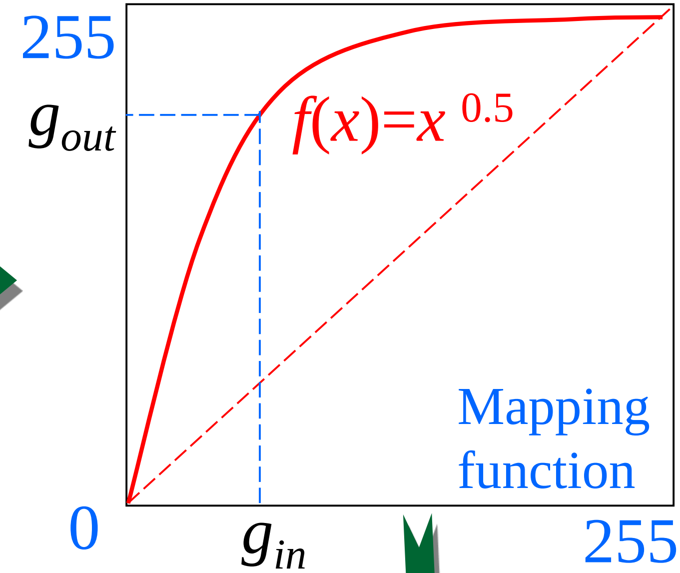


Mapping function (Reverse)

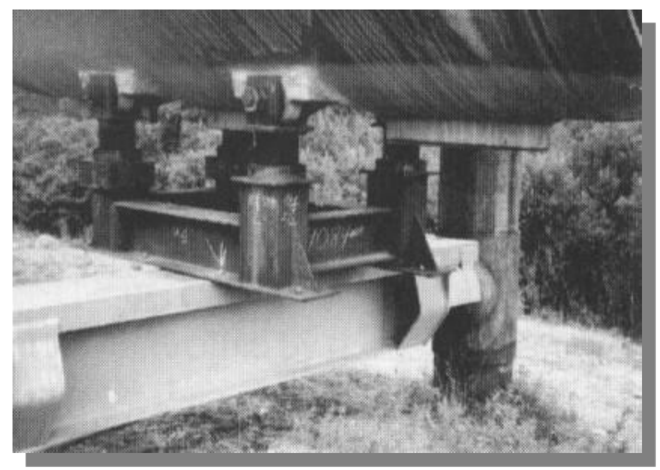


$$f(x) = 255 - x$$

Input image



Intensity mapping using the function $f(x) = x^{0.5}$ (Gamma correction) boosts dark pixels much more than bright ones

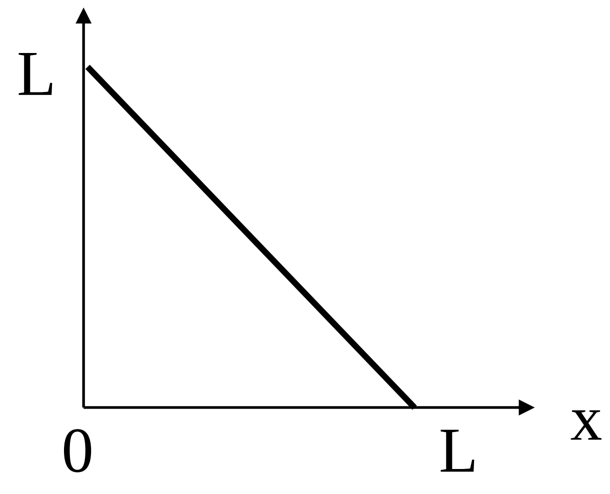


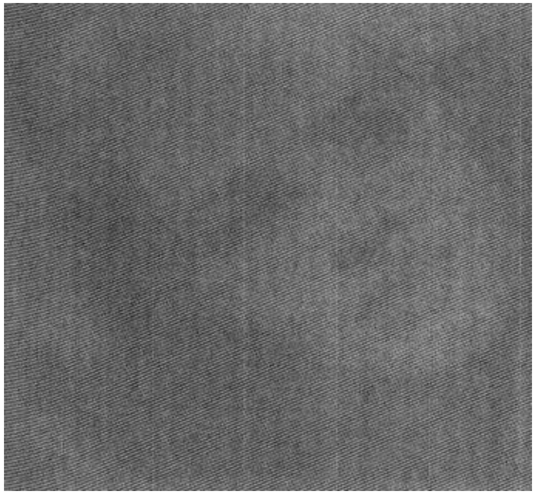
Output image

Original, fairly dark, Alaskan Pipeline image

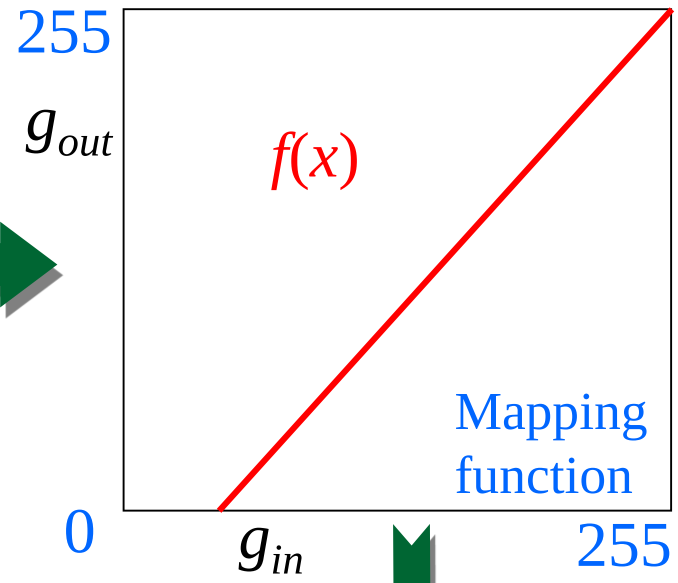
Digital Negative

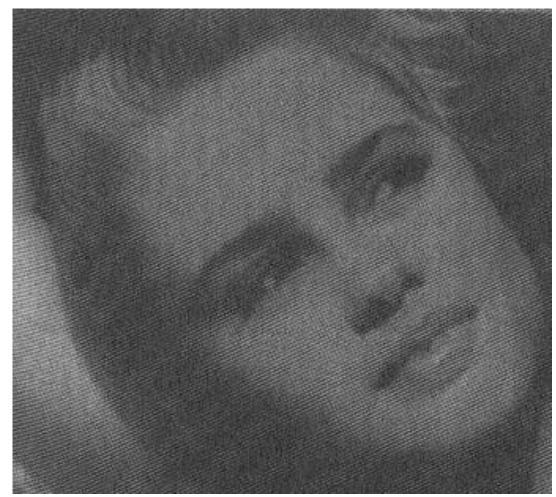
$$y = L - x$$





Input image

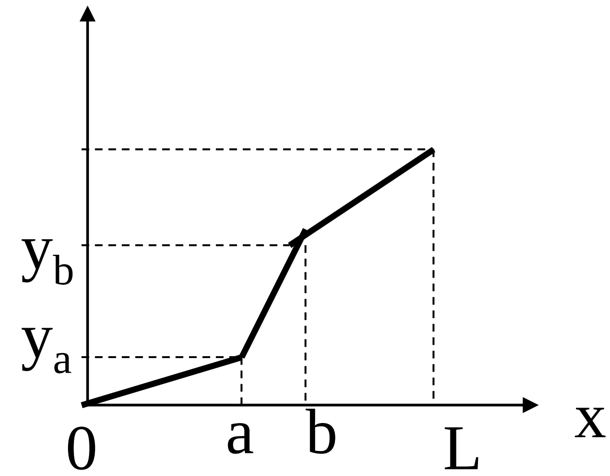


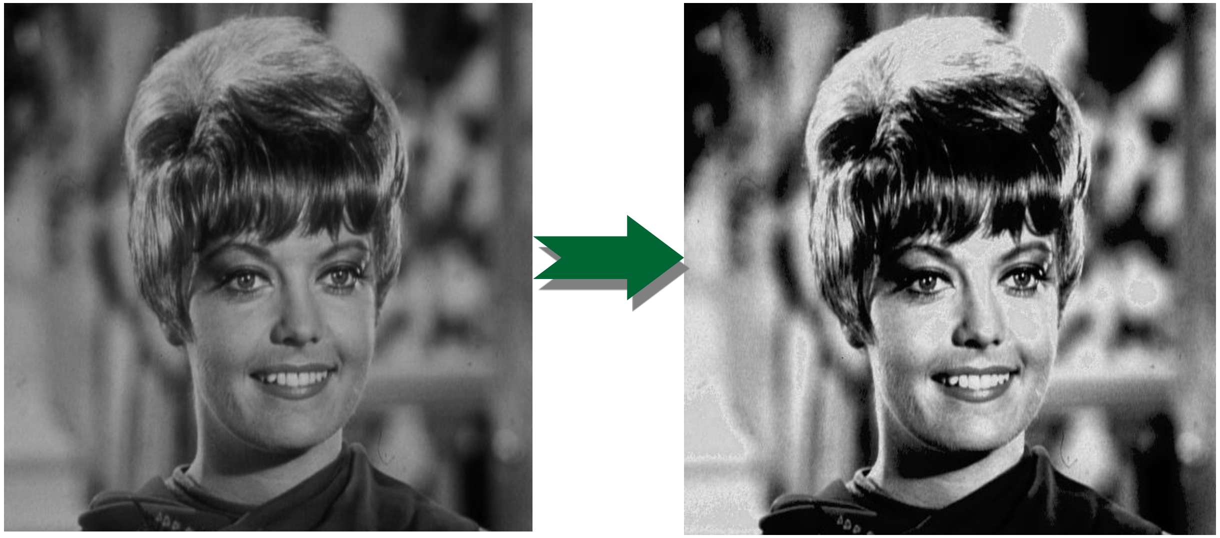


Output image

Contrast Stretching

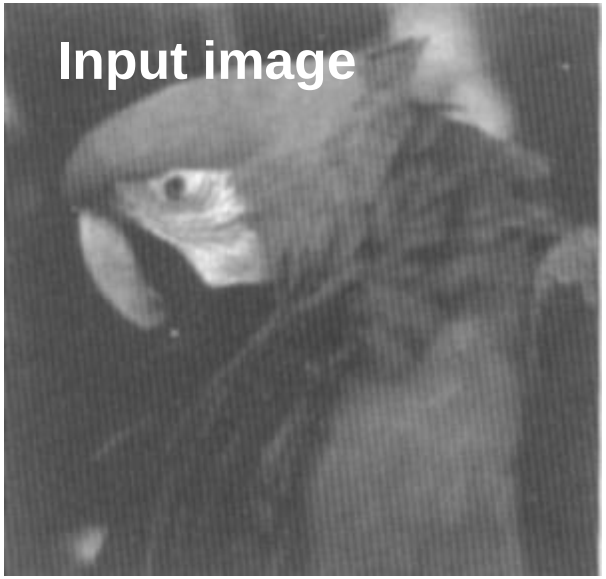
$$y = \begin{cases} \alpha x & 0 \le x < a \\ \beta(x-a) + y_a & a \le x < b \\ \gamma(x-b) + y_b & b \le x < L \end{cases}$$

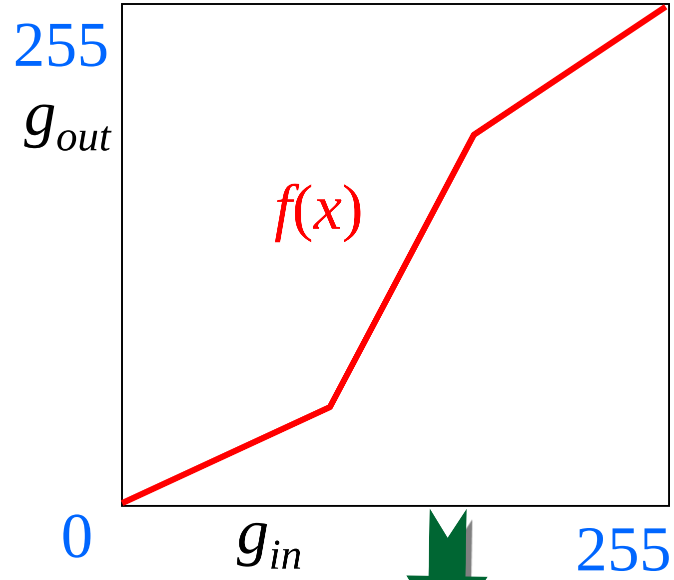


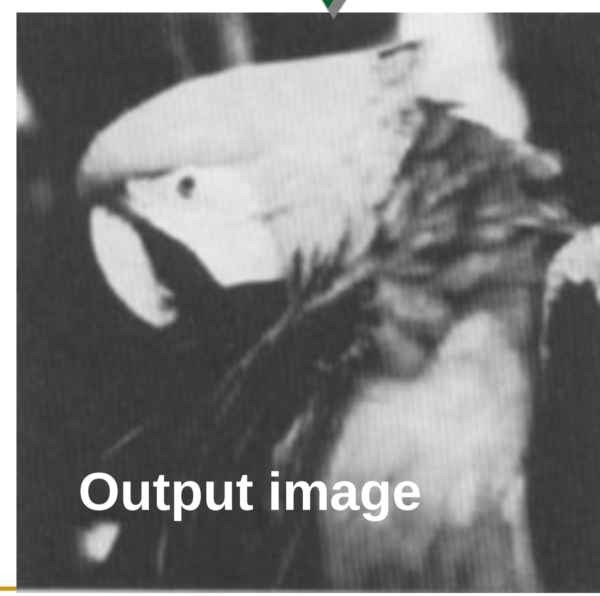


$$a = 50, b = 150, \alpha = 0.2, \beta = 2, \gamma = 1, y_a = 30, y_b = 200$$

Mapping function

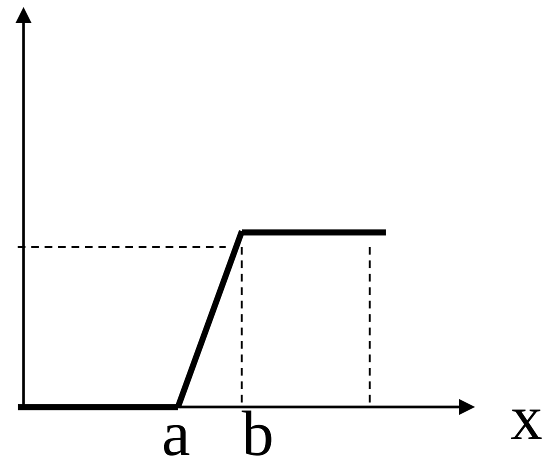






Clipping

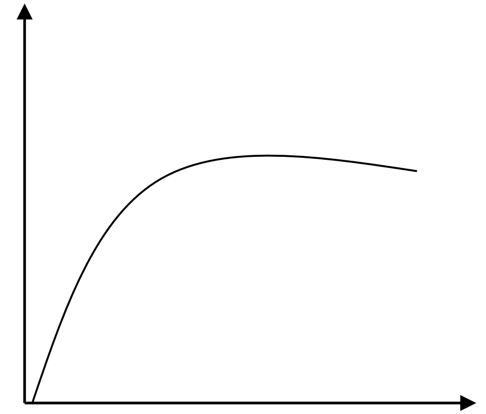
$$y = \begin{cases} 0 & 0 \le x < a \\ \beta(x-a) & a \le x < b \\ \beta(b-a) & b \le x < L \end{cases}$$



$a = 50, b = 150, \beta = 2$

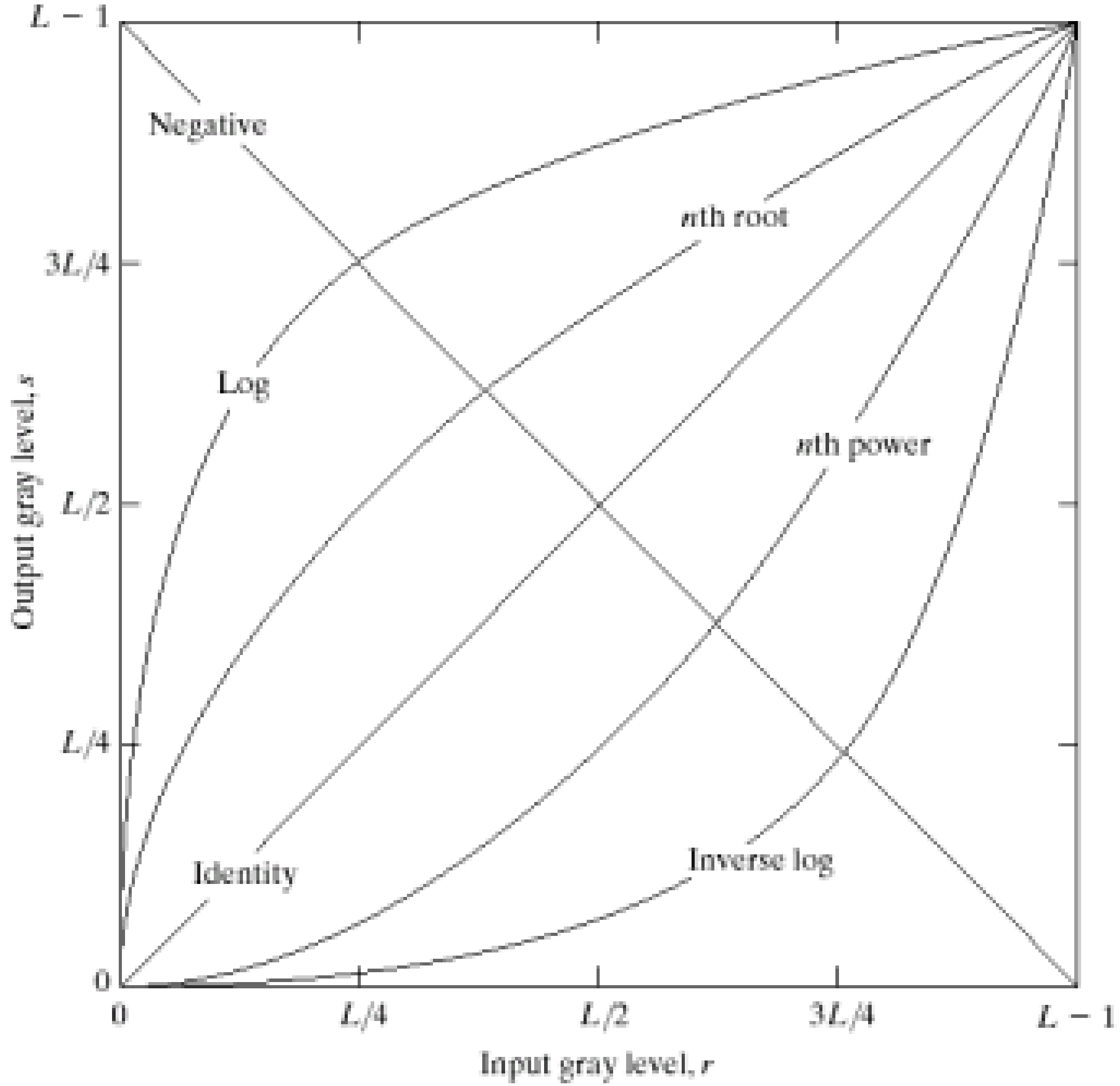
Range Compression

$$y = c\log_{10}(1+x)$$



c=100

FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



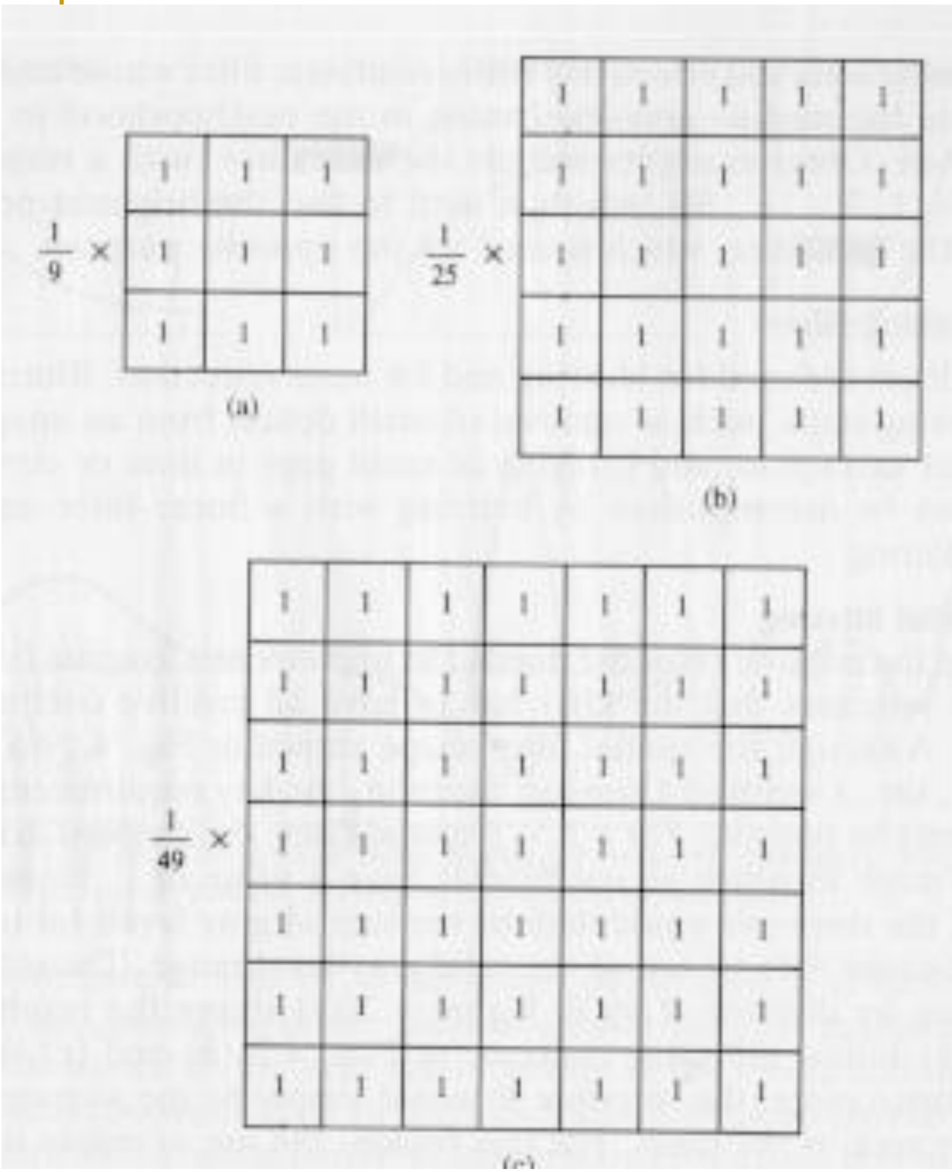
Spatial Correlation and Convolution(相關與卷積）

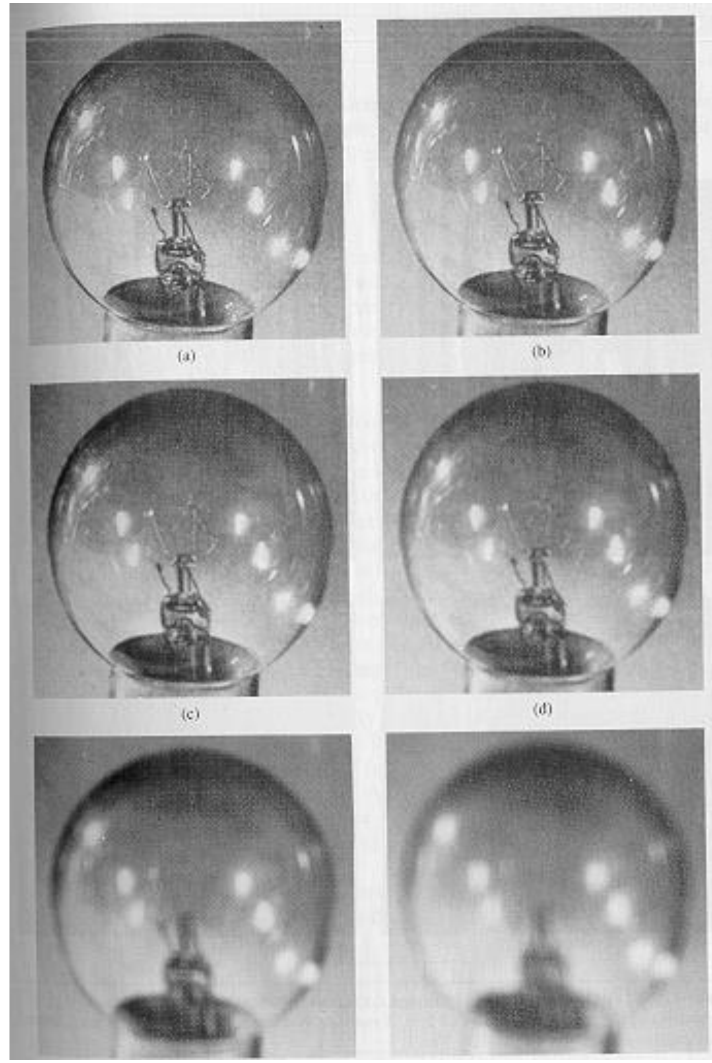
- **Correlation**: the process of moving a filter mask over the image and computing the sum of products at each location.
- **Convolution**: the filter is first rotated by 180°
    - Each pixel in filter ω visits every pixel in image *f*
    - If there are parts of the functions that do not overlap, the solution is to pad *f* with enough 0s on each side to allow each pixel in ω to visit every pixel in *f*

Smoothing (or Low-pass) filters

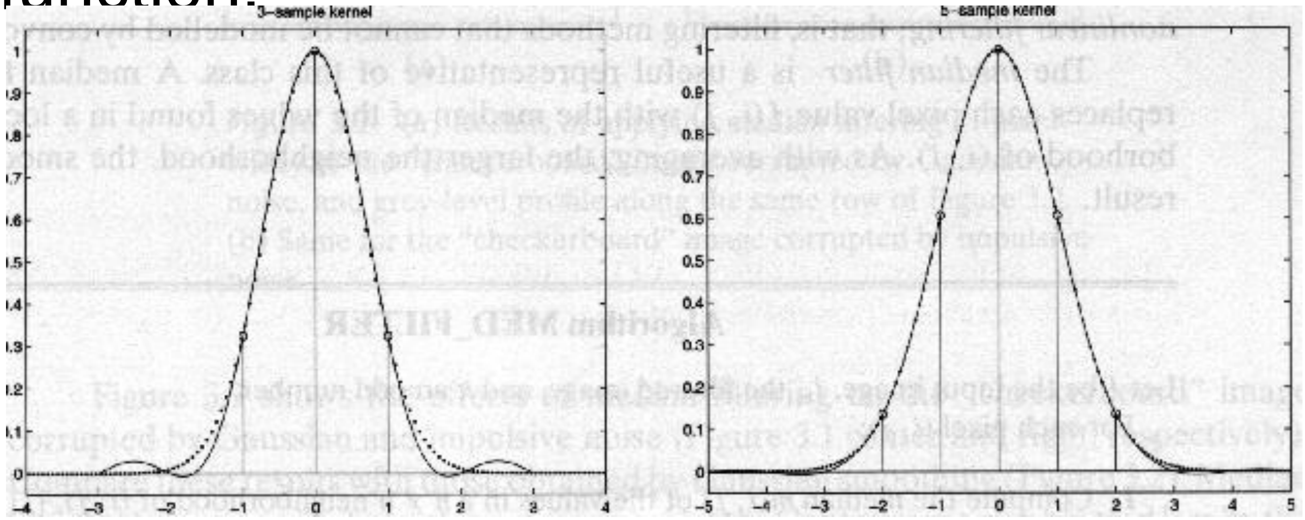
- Useful for noise reduction and image blurring.
- It removes the finer details of an image.

- **Averaging or Mean filter**
  - The elements of the mask must be positive.
  - The size of the mask determines the degree of smoothing.

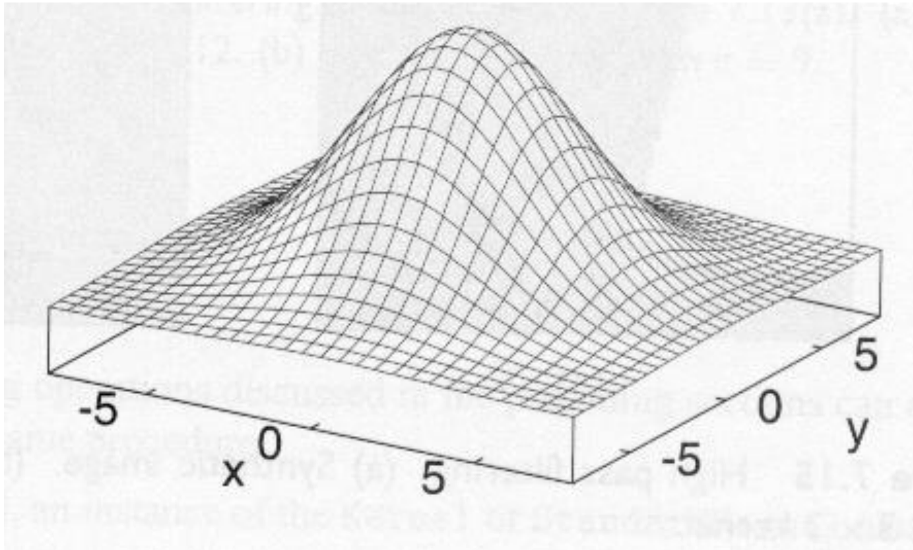

$\frac{1}{9}\times$
(a)
$\frac{1}{25}\times$
(b)
$\frac{1}{49}\times$
(c)


(a)
(b)
(c)
(d)

- **Gaussian (linear filter)**
  - The weights are samples from a Gaussian function.



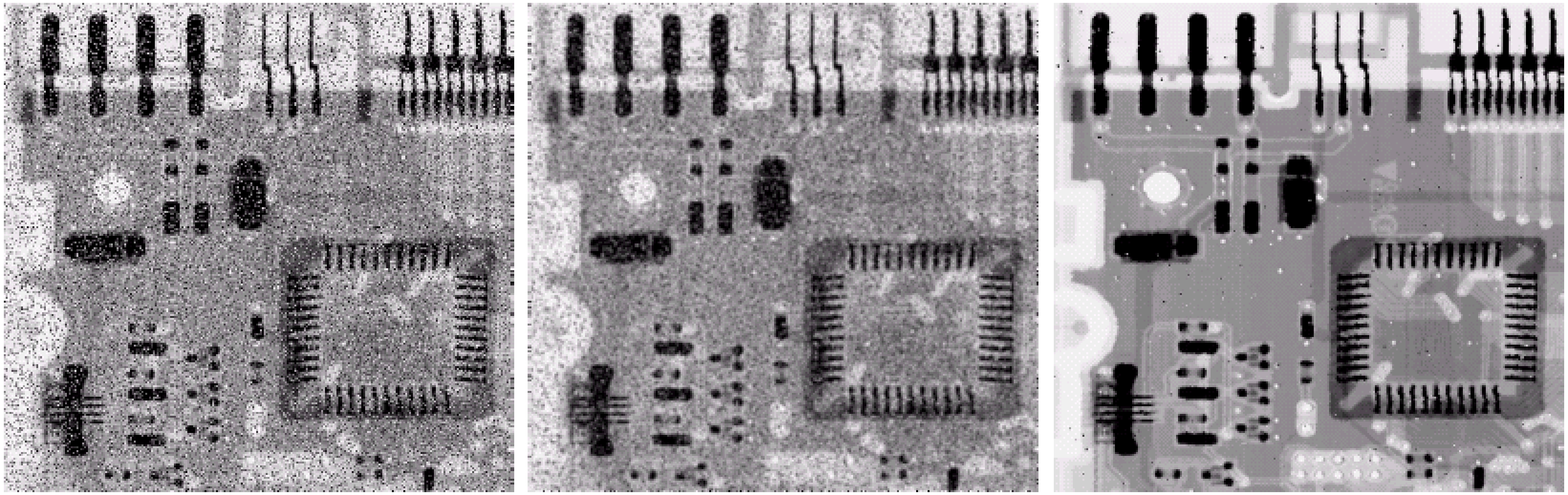
(example using 1D Gaussian)



$$h(x, y) = e^{\left[\frac{-(x^2+y^2)}{2\sigma^2}\right]}$$

Median filter (non-linear)

- Effective for removing "salt and pepper" noise (random occurrences of black and white pixels).



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

- Advantage
  - Provide excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filter of similar size

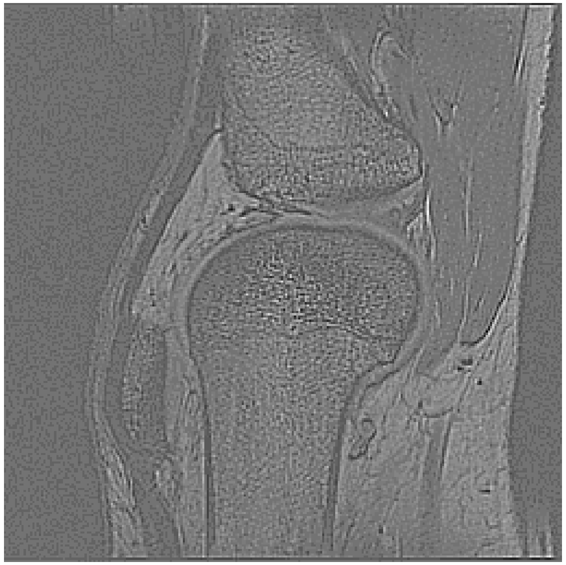
Sharpening (or High-pass)

- It is used to emphasize the fine details of an image (has the opposite effect of smoothing).
- Points of high contrast can be detected by computing intensity **differences** in local image regions.
- The weights of the mask are both positive and negative.

- Sharpening an image increases the contrast between bright and dark regions to bring out features.
- The sharpening process is basically the application of a high pass filter to an image. The following array is a kernel for a common high pass filter used to sharpen an image

$$\begin{bmatrix} -1/8 & -1/8 & -1/8 \\ -1/8 & 1 & -1/8 \\ -1/8 & -1/8 & -1/8 \end{bmatrix}$$

Original



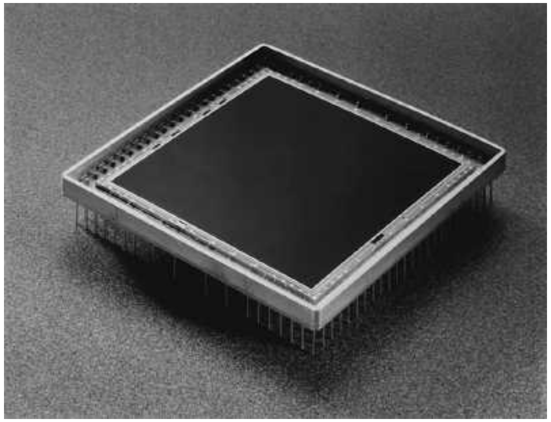
Sharpen Filtered：Pixels that differ dramatically in contrast with surrounding pixels are brightened

Unsharp masking and highboost filtering

- Unsharp masking: subtracting an unsharp (smoothed) version of an image from the original image
  - Blur the original image
  - Subtract the blurred image from the original (the resulting difference is called the mask)
  - Add the mask to the original

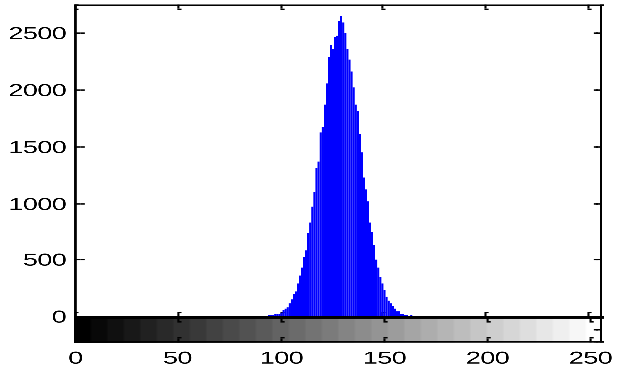
Noise

- Source of noise = CCD chip.
- Electronic signal fluctuations in detector.
- Caused by thermal energy.
- Worse for infra-red sensors.



Noise Histogram

- Plot noise histogram
- Histogram is called normal or Gaussian (distribution)
- Mean (noise) $\mu = 0$
- Standard deviation σ
- *i* is the grey level.



$$f(i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{i-\mu}{\sigma}\right)^2\right]$$

- **Gaussian noise** is statistical noise that has its probability density function equal to that of the normal distribution, which is also known as the Gaussian distribution.
- A special case is **white Gaussian noise**, in which the values at any pairs of times are statistically independent (and uncorrelated).
- In applications, Gaussian noise is most commonly used as additive white noise to yield additive white Gaussian noise

- **Salt and pepper noise** is a form of noise typically seen on images.
- It represents itself as randomly occurring white and black pixels.

- The **grain** of photographic film is a signal-dependent noise, is uniformly distributed (equal number per area)

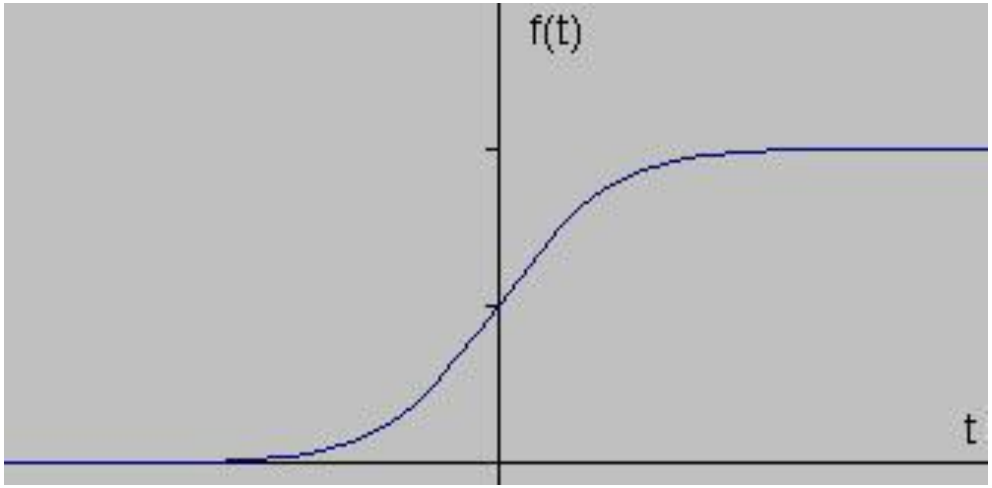
Edge Detection

- Detecting Edges in an image **significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image.**

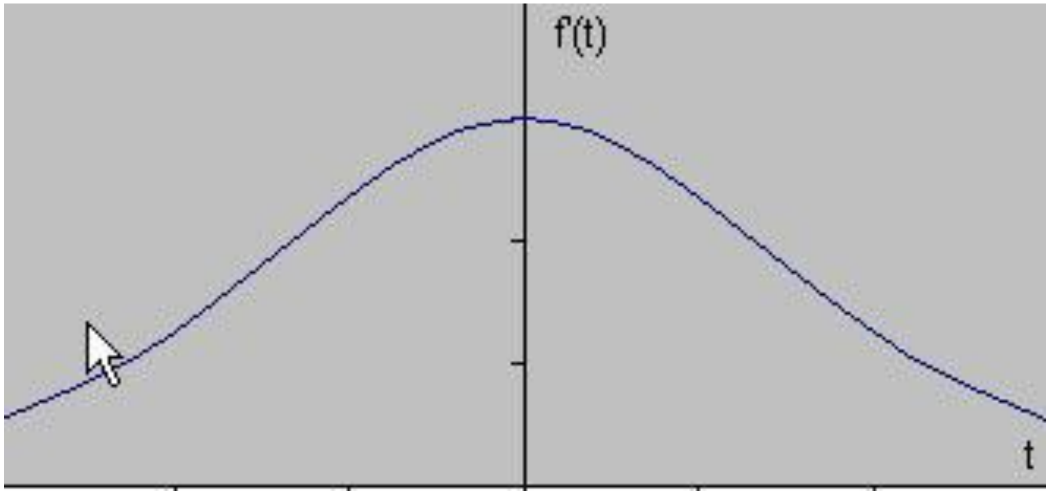
- There are many ways to perform edge detection. However, the majority of different methods may be grouped into two categories,
  - Gradient （梯度）
  - Laplacian

- The ***gradient method*** detects the edges by looking for the maximum and minimum in the first derivative of the image.
- The ***Laplacian method*** searches for zero crossings in the second derivative of the image to find edges.

- An edge has the one-dimensional shape of a ramp and calculating the derivative of the image can highlight its location.
- Suppose we have the following signal, with an edge shown by the jump in intensity below:

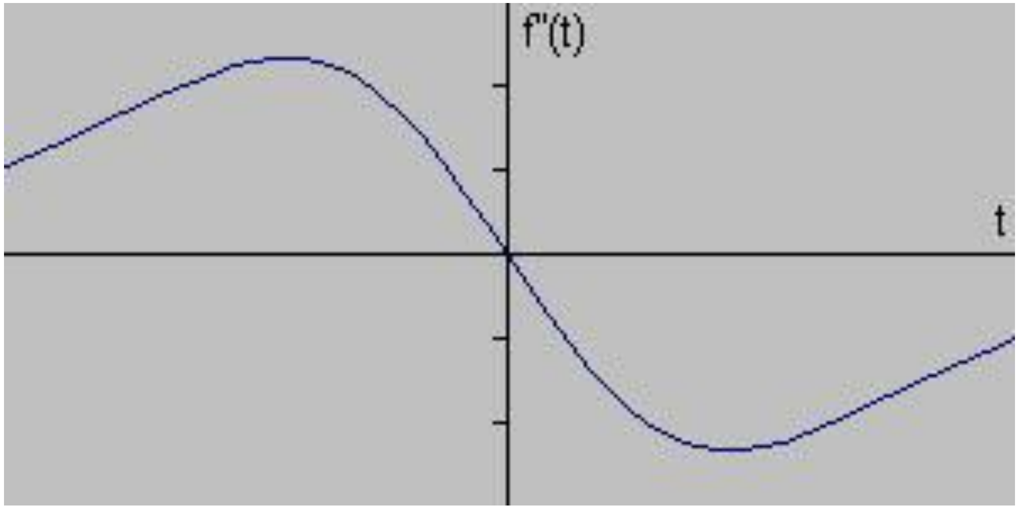


- If we take the gradient of this signal (which, in one dimension, is just the first derivative with respect to t) we get the following:



- The derivative shows a maximum (changing rate) located at the center of the edge in the original signal.

- When the first derivative is at a maximum, the second derivative is zero.
- Another alternative to finding the location of an edge is to locate the zeros in the second derivative.



Using masks to estimate the gradient

- ## Sobel edge detection
- **Uses templates in the form of convolution masks having the following values**

$$s_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

$$s_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

- The magnitude of the gradient is then calculated using the formula:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

- An approximate magnitude can be calculated using:

$$|G| = |Gx| + |Gy|$$

Laplacian Edge Detector

- The 5x5 Laplacian used is a convoluted mask to approximate the second derivative, unlike the Sobel method which approximates the gradient.
- Instead of 2 3x3 Sobel masks, one for the x and y direction, Laplace uses 1 5x5 mask for the 2nd derivative in both the x and y directions.

- Edge detection by Laplace operator followed by **zero-crossing detection**
- If in the neighborhood (3x3, 5x5, 7x7, etc.) of a given pixel there exist both polarities, i.e., pixel values greater than and smaller than 0, then the pixel is a zero-crossing. (Note that the pixel in question does not have to be zero.)
- Specifically, we find the maximum and minimum among all pixels in the neighborhood of a pixel under consideration. If the maximum is greater than zero and the minimum is smaller than zero, the pixel is a zero-crossing.

Algorithm

- Apply Laplace mask to the whole image.
- Zero-crossing process.
  - Find the maximum and minimum among all pixels in the neighborhood of a pixel.
  - Using either 1 or 2 for the zero-crossing process
  1. If the maximum is greater than zero and the minimum is smaller than zero, the pixel is a zero-crossing.
  2. If the difference between the maximum and the minimum is greater than a threshold value, the pixel is on an edge.

Canny edge detector

- In 1986, John Canny defined a set of goals for an edge detector and described an optimal method for achieving them

Canny edge detector algorithm

1. The first step is to filter out any noise in the original image before trying to locate and detect any edges.
   - **The larger the width of the Gaussian mask, the lower is the detector's sensitivity to noise**.
   - The localization error in the detected edges also increases slightly as the Gaussian width is increased.

2. After smoothing the image and eliminating the noise, the next step is to find the edge strength by taking the gradient of the image.
   - The Sobel operator performs a 2-D spatial gradient measurement on an image.
   - Then, the approximate absolute gradient magnitude (edge strength) at each point can be found.

3. **Finding the edge direction is trivial once the gradient in the x and y directions are known. However, G_x *generates an error whenever it is equal to zero*.**
   - So in the code there has to be a restriction set whenever this takes place. Whenever the gradient in the x direction is equal to zero, the edge direction has to be equal to 90 degrees or 0 degrees, depending on what the value of the gradient in the y-direction is equal to.
   - If **Gy has a value of zero, the edge direction will equal 0 degrees**. **Otherwise the edge direction will equal 90 degrees**. The formula for finding the edge direction is just:

$$\theta = arctg(G_y / G_x)$$

4. Once the edge direction is known, the next step is to relate the edge direction to a direction that can be traced in an image. So if the pixels of a 5x5 image are aligned as follows:

x	x	x	x	x
x	x	x	x	x
x	x	a	x	x
x	x	x	x	x
x	x	x	x	x

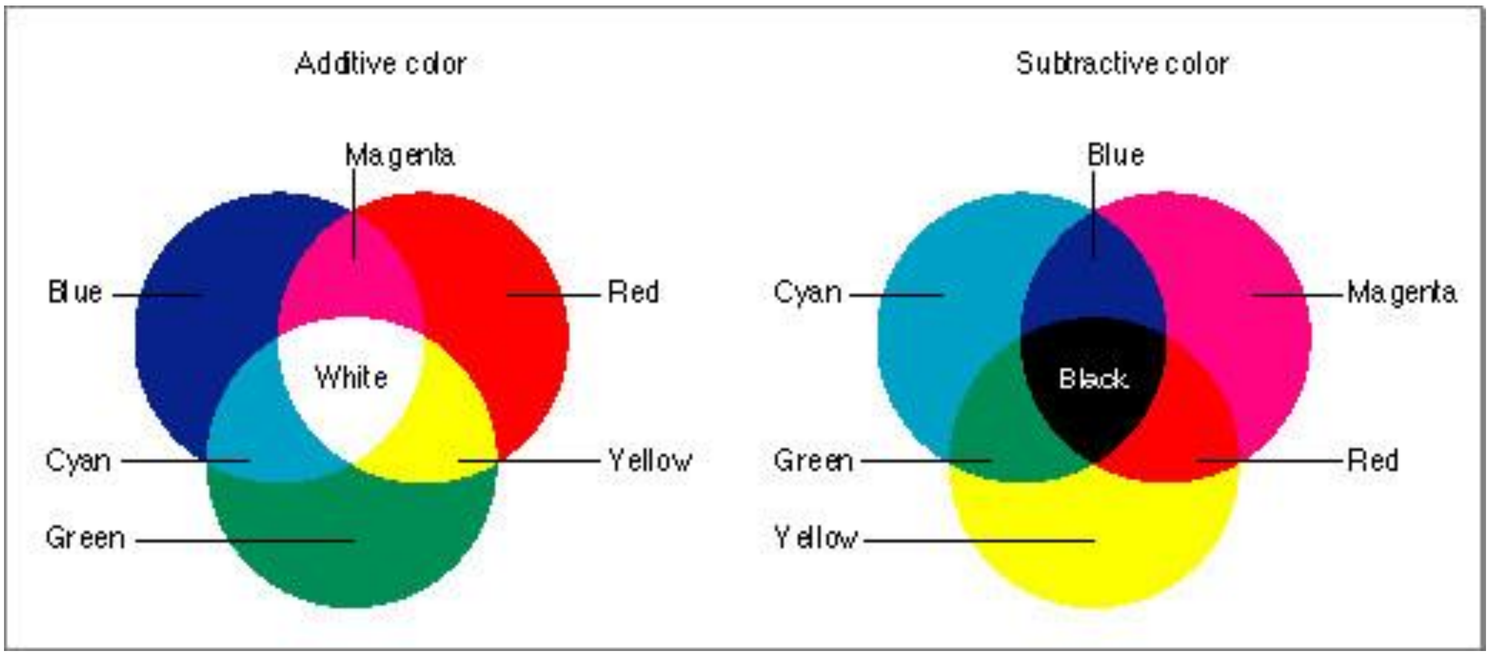
5. After the edge directions are known, nonmaximum suppression now has to be applied. Nonmaximum suppression is used to trace along the edge in the gradiant direction and suppress any pixel value (sets it equal to 0) that is not considered to be an edge. This will give a thin line in the output image.
6. Finally, hysteresis is used as a means of eliminating streaking.
   - Streaking is the breaking up of an edge contour caused by the operator output fluctuating above and below the threshold.

Color Models

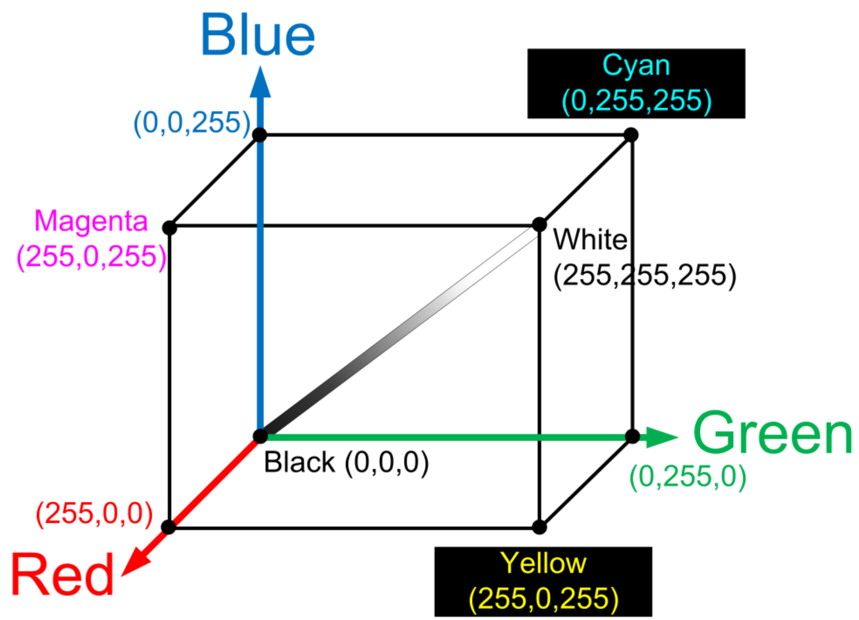
- The purpose of a color model (also called color space or color system) is to facilitate the specification of colors in some standard, generally accept way.
- **RGB** (red,green,blue) : monitor, video camera.
- **CMY**(cyan,magenta,yellow),**CMYK** (CMY, black) model for color printing.
- and **HSI** model, which corresponds closely with the way humans describe and interpret color.

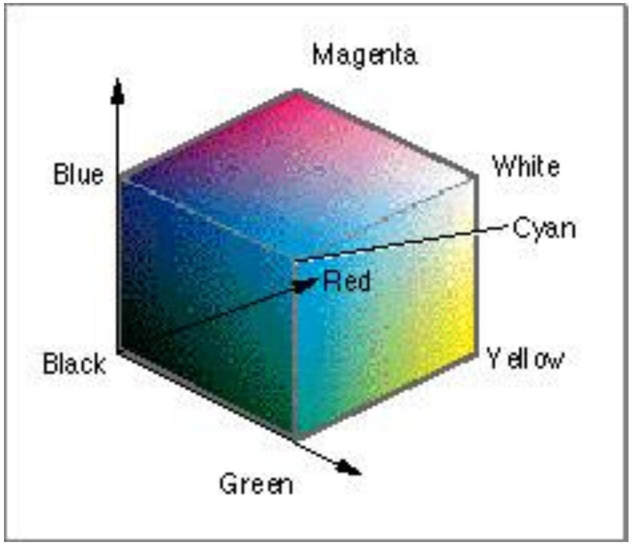
R+G+B=White? Really?!?

- So why don't we get white, when we use paint? *Subtractive Color!*
- But why does it work for the TV? *Additive Color!*



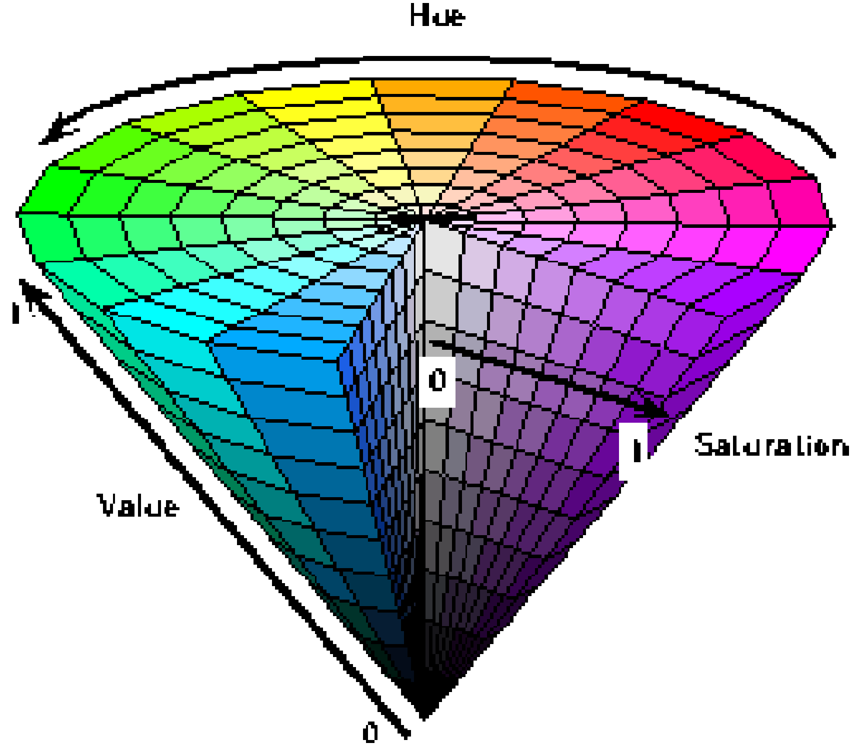
RGB color space





HSI color space

- Perhaps the most intuitive color representation!



RGB→HSI

$$I = \frac{1}{3}(R + G + B)$$

$$S = 1 - \frac{3}{(R + G + B)}[\min(R, G, B)]$$

$$H = (\{ \begin{matrix} \theta & G \geq B \\ 2\pi - \theta & G < B \end{matrix})/(2\pi)$$

$$\theta = \arccos\{\frac{\frac{1}{2}[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{\frac{1}{2}}}\}$$

$$If\ S = 0, H = 0$$

HSI→RGB

$$(1)\ H \in [0, 2\pi/3]$$

$$B = I(1 - S)$$

$$R = I[1 + \frac{s\cos H}{\cos(\frac{\pi}{3} - H)}]$$

$$G = 3I - (B + R)$$

$$(2)\ H \in [2\pi/3, 4\pi/3]$$

$$R = I(1-S)$$

$$G = I[1+\frac{s\cos(H-\frac{2\pi}{3})}{\cos(\pi-H)}]$$

$$B = 3I-(R+G)$$

$$(3)\ H \in [4\pi/3, 2\pi]$$

$$G = I(1 - S)$$

$$B = I[1 + \frac{s\cos(H - \frac{4}{3}\pi)}{\cos(\frac{5\pi}{3} - H)}]$$

$$R = 3I - (G + B)$$

Different usage of RGB and HSI color models.

- HSI Models is especially useful for image processing on the isolated intensity component (preserves color).
- RGB is typically used for computer graphics.

Histogram equalization Algorithm

- convert from RGB to HSI model
- get intensity
- Histogram equalization
- convert from HSI to RGB model
- Output RGB file

Segmentation in HSI space

- S: as a template, determine the value of the threshold:
  - S(i,j)=1, if S(I,j)>threshold
  - S(i,j)=0, if S(I,j)<threshold
- H: do the thresholding in H: S*H
- I: there is no any color information.

Segmentation in RGB space

- **Find the color we would like to segment: a**
- **To every pixel in image (the color difference is small)**
- **If**

$$D(z,a) = \| z - a \| = [(z_R - a_R)^2 + (z_G - a_G)^2 + (z_B - a_B)^2]^{\frac{1}{2}} \leq D_0$$

- **Then I(R,G,B)=1, otherwise I(R,G,B)=0**

Full-Color Image Processing
Color Image Smoothing

Averaging:

$$\bar{\mathrm{c}}(x,y)=\frac{1}{K}\sum_{(x,y)\in S_{xy}}\mathrm{c}(x,y)$$

$$\bar{\mathrm{c}}(x,y)=\begin{bmatrix}\dfrac{1}{K}\sum_{(x,y)\in S_{xy}}R(x,y)\\ \dfrac{1}{K}\sum_{(x,y)\in S_{xy}}G(x,y)\\ \dfrac{1}{K}\sum_{(x,y)\in S_{xy}}B(x,y)\end{bmatrix}$$

Full-Color Image Processing
Color Image Sharpening

The Laplacian of Vector **c** :

$$\nabla^2\left[\mathbf{c}(x,y)\right]=\begin{bmatrix}\nabla^2 R(x,y)\\ \nabla^2 G(x,y)\\ \nabla^2 B(x,y)\end{bmatrix}$$

Fourier Transform and Applications

- Fourier was obsessed with the physics of heat and developed the Fourier transform theory to model heat-flow problems
- **Crazy idea**

Any *univariate (單因素) function can be rewritten as a weighted sum of sines and cosines of different frequencies*

Joseph Fourier (1768-1830)

Frequency Domain Filtering

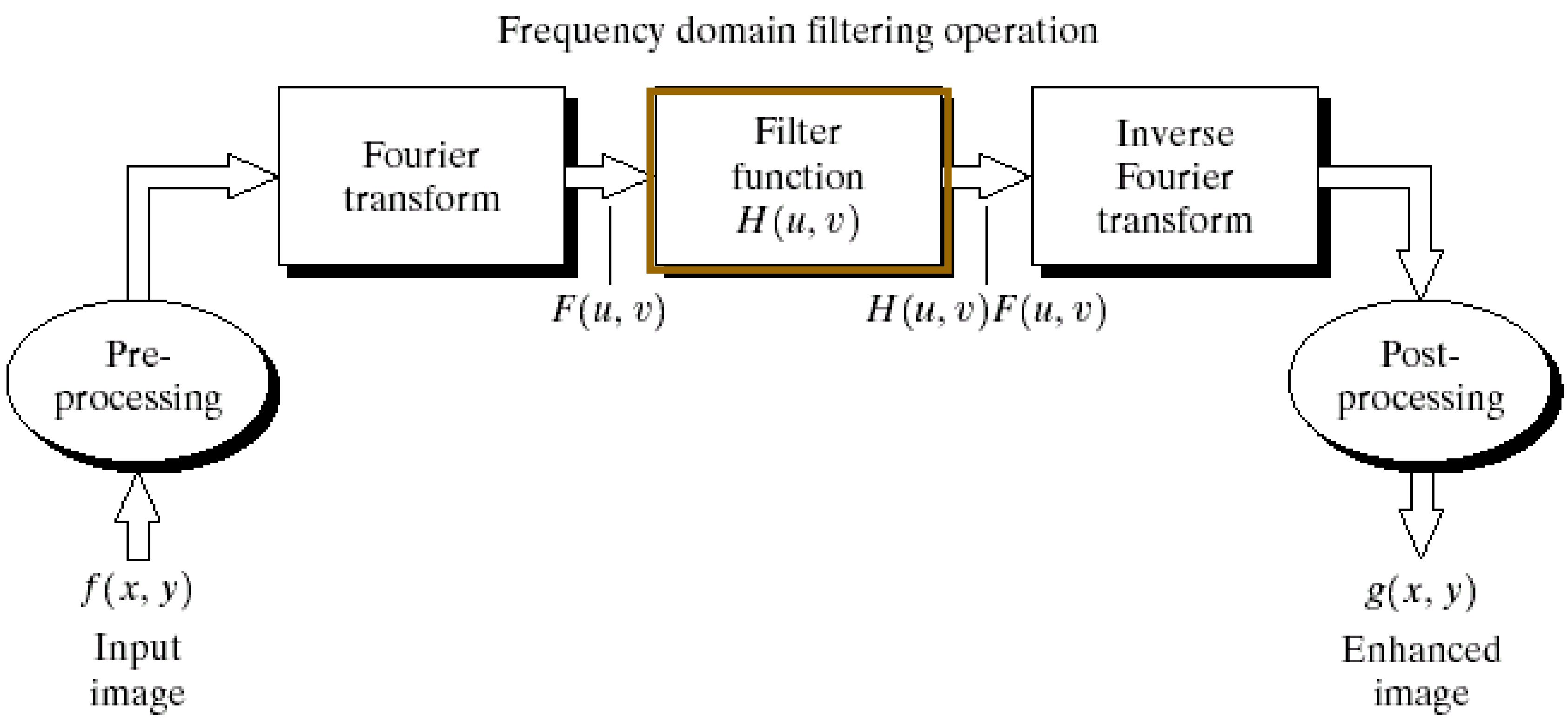


FIGURE 4.5 Basic steps for filtering in the frequency domain.

Frequency Domain Filtering

- Edges and sharp transitions (e.g., noise) in an image contribute significantly to high-frequency content of FT.
- Low frequency contents in the FT are responsible to the general appearance of the image over smooth areas.
- Blurring (smoothing) is achieved by attenuating range of high frequency components of FT.