Graphics & Multimedia

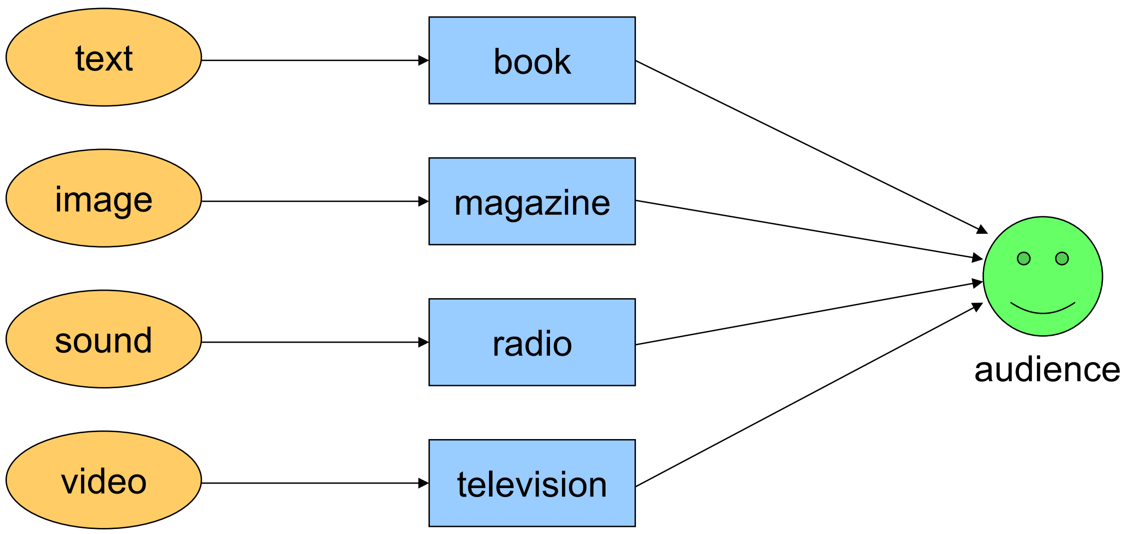
School of Information Technologies
The University of Sydney

Week01 • Semester 2 • 2018

Introduction / Image Representation & Enhancement

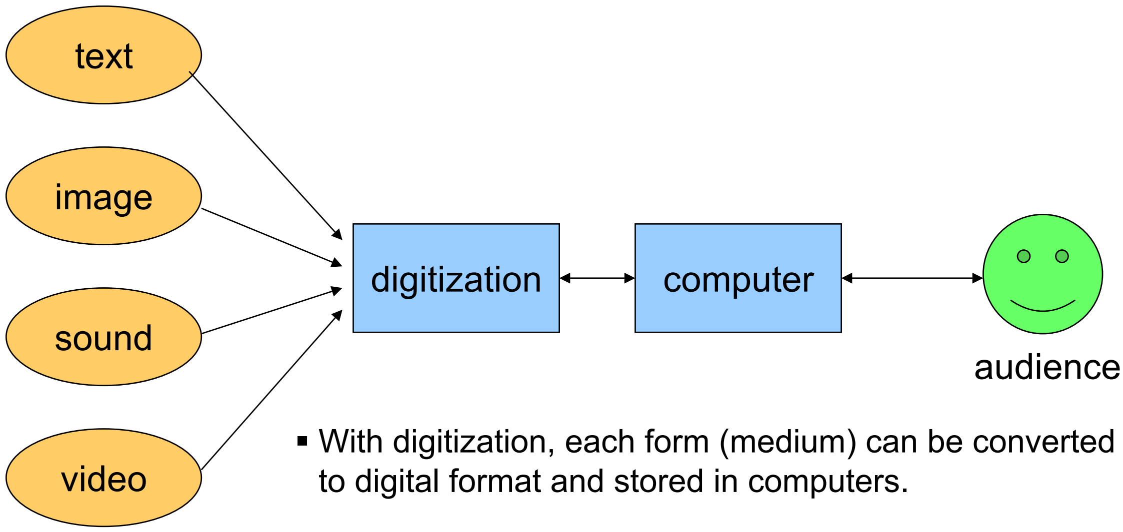
- Introduction / Course overview / Administrative issues
- Enabling technologies
  - Digitization / Digital data acquisition / Visual perception
- Representing digital image
- Spatial and gray-level resolution
- Spatial domain image processing
- Image enhancement
  - Contrast stretching
  - Histogram equalization
  - Noise smoothing and filtering

Multimedia before the Digital Era



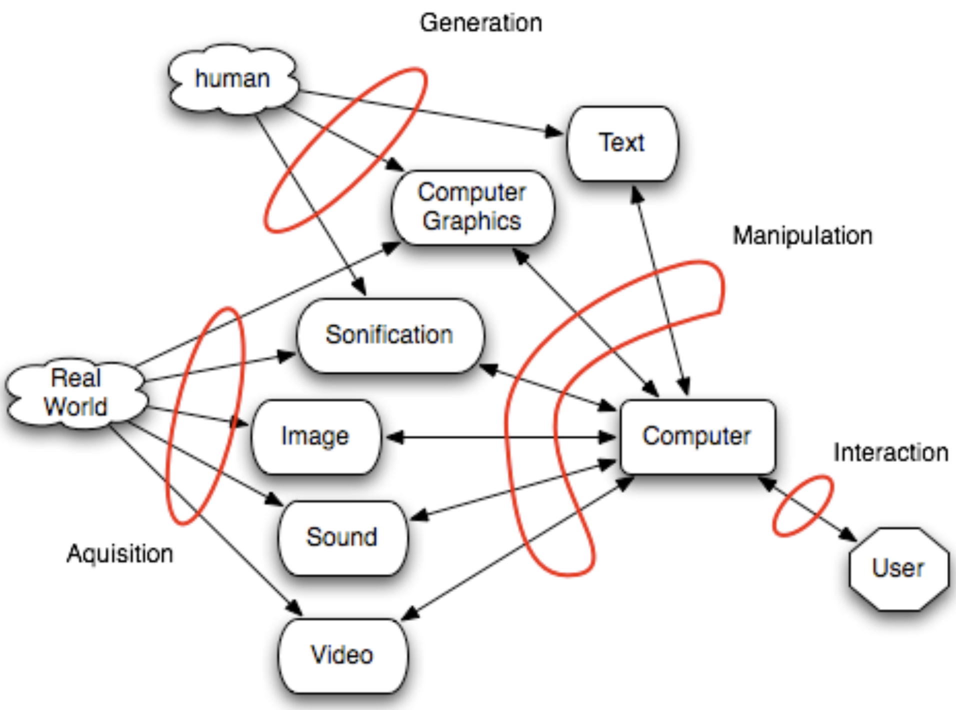
Before digitization, each form (medium) of human communication has its own technology and its own channel.

Multimedia in the Digital Era



- With digitization, each form (medium) can be converted to digital format and stored in computers.
- It allows us to handle all forms of human communication with equal facility; to combine several forms into a single multimedia work; and to allow **interactivity** between the viewer and the author.

Digital Multimedia



Digital Multimedia and Applications

- **Digital multimedia** is the computer-controlled integration of multiple media sources e.g., text, graphics, images, sound / audio, animation, and video where every type of media can be represented, stored, transmitted and processed digitally.
- Multimedia has a wide range of application, from entertainment to education and business: computer games, special effects, virtual reality training, scientific visualization, digital health, public security …
- Job opportunities: more and more people with multimedia computing / development skills will be necessary for the creation and maintenance of various multimedia applications.

Course Overview

This unit provides a broad introduction to the field of graphics and multimedia computing to meet the diverse requirements of application areas such as entertainment, industrial design, virtual / augmented reality, vision-based multimedia analytics, digital healthcare, intelligent media management, social media and remote sensing. It covers both the underpinning theories and the practices of computing and manipulating digital media including graphics / image, audio, animation, and video. Emphasis is placed on principles and cutting-edge techniques for multimedia data processing, content analysis, media retouching, media coding and compression.

- Graphics / Image Data Representation & Enhancement
- Morphological Image Processing
- Color Models & Color Image Processing
- Video Data Representation & Processing
- 2D / 3D Computer Graphics
- Computer Animation & Multimedia Authoring
- Digital Audio Processing and Coding
- Image & Video Coding and Compression
- Advanced Multimedia Computing

UoS Administrative Issues

- Lectures / lecture notes
- Lab / tutorials
- eLearning website / resources
- Assessment

Tasks	Introduced	Due	Marks
Lab Activities	Week2	Week12	15%
Assignment (individual)	Week2	Week12	25%
Final Examination *			60%

** Final examination preparation (Week13)*

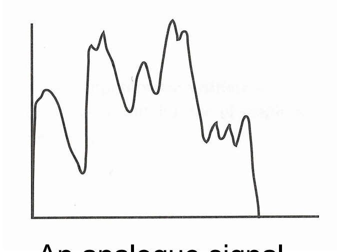
Image Representation and Enhancement

- Enabling technologies
  - Digitization
  - Digital Data Acquisition
  - Visual Perception
- Representing digital image
- Spatial and gray-level resolution
- Spatial domain image processing
- Image enhancement
  - Contrast stretching
  - Histogram equalization
  - Noise smoothing and filtering

Digitization

- All types of media are represented digitally as patterns of bits.

- **Observation:** when we have a continuously varying signal, both the **value** we measure, and the **intervals** at which we can measure it, can vary infinitesimally.



An analogue signal

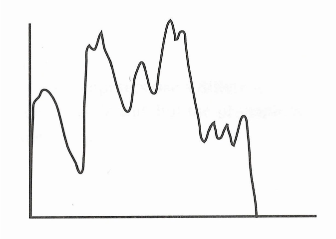
- In contrast, if we were to convert it to a digital signal, we would have to restrict both of these to a set of discrete values. That is, *digitization* – which is what we call the process of converting a signal from analogue to digital form.

Digitization

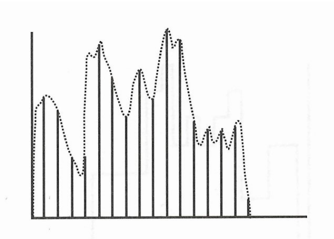
- Digitization consists of two steps:

- *Sampling*, when we measure the signal's value at discrete intervals;
- *Quantization*, when we restrict the value to a fixed set of levels.

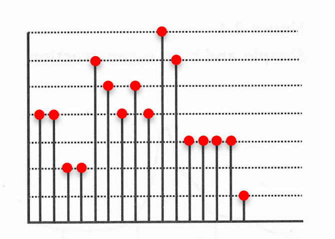
- Sampling and quantization can be carried out in either order, by special hardware devices – *analogue to digital converters (ADCs)*.



An analogue signal



Sampling

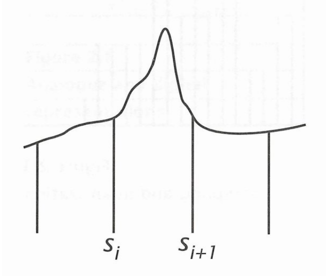


Quantization

Digitization

- The number of samples in a fixed amount of time or space is known as the *sampling rate*.

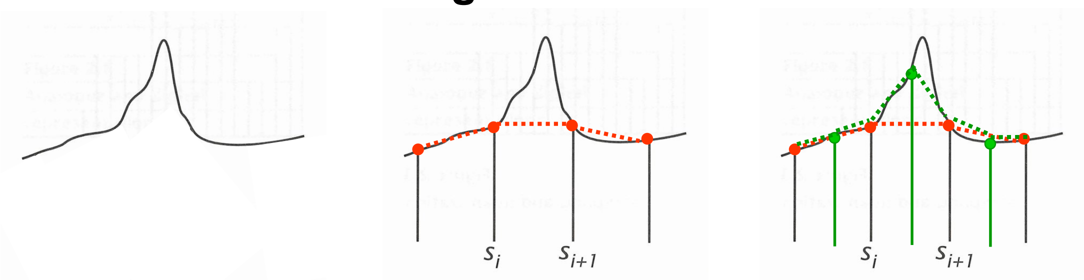
- **Observation:** The values of the consecutive samples taken at S_i and S_{i+1} are identical, and there cannot possibly by any way of inferring from them the presence of the spike in between those two points – the signal could as easily have dipped down or stayed at the same level.



Under-sampling

- The effects of such *under-sampling* on the way in which the reconstructed signal will be perceived depend on what the signal represents – sound, image and so on – and whether it is time-varying or space-varying.

Digitization



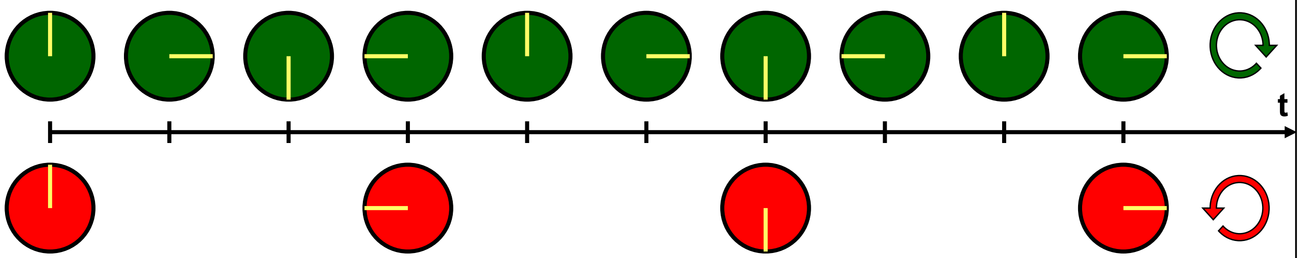
- It is easy enough to see that if the sampling rate is too low, some detail will be lost in the sampling.

- It is less easy to see whether there is ever any rate at which we can be sure samples are close enough together to allow the signal to be accurately reconstructed, and if there is, how close is close enough.

- The *Shannon Sampling Theorems* – if the highest frequency component of a signal is at f_h , the signal can be properly reconstructed if it has been sampled at a frequency greater than $2f_h$. This limiting value is known as the *Nyquist rate*.

Example: Suppose we have a circular disk, with a single radial line marked on it, which is spinning in a clockwise direction at a rate of n rotations per second, and suppose that we "sample" this rotation as a movie camera would, by taking snapshots of the disk at equal time intervals.

Sampling 4n times a second – Considered as a periodic signal, the rotating disk has a frequency of n and we have sampled at 4n, comfortably above the Nyquist rate.



Sampling at a rate of 4n/3 – It appears from the successive positions of the line more as if the disk is rotating anti-clockwise at a rate of 4n/3.

* Note that we must go beyond the Nyquist rate: if we sample our disk at a rate of exactly 2n, we get samples in which the line alternates between the 12 o'clock and 6 o'clock positions, so that it is impossible to determine whether the disk is rotating clockwise or anti-clockwise.

The phenomenon may be familiar to you from Western movies, in which stagecoach wheels frequently appear to rotate backwards, because the rate at which the film frames were shot is less than the Nyquist rate relative to the actual rotational speed of the wheels.

Digitization

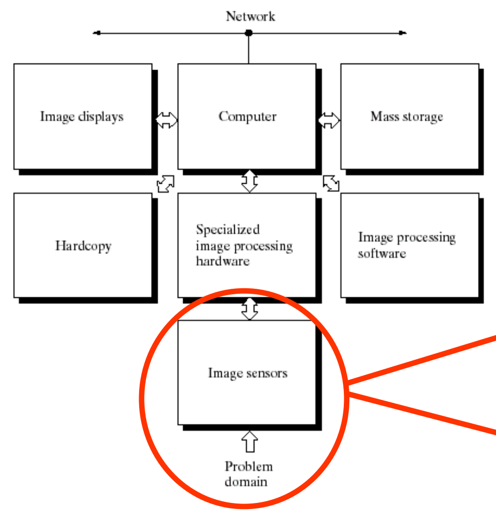
- In general, if we under-sample a signal – sample it at less than the Nyquist rate – some frequency components in the original will get transformed into other frequencies when the signal is reconstructed.
- This phenomenon is known as *aliasing*, and is perceived in different ways in different media:
  - In sound, it is heard as distortion;
  - In images, it is usually seen in the form of jagged edges, or, where the image contains fine repeating details (e.g., *Moire patterns*);
  - In moving pictures, temporal under-sampling leads to jerkiness of motion, as well as phenomena similar to the retrograde disk just described.

- The human ear is generally considered to be able to detect frequencies in the range between 20 Hz and 20 kHz

- If the limit of hearing is taken to be 20kHz, a minimum sampling rate of 40 kHz is required by the Sampling Theorem.
  - The sampling rate used for audio CDs is 44.1kHz – the precise figure being chosen by manufacturers to produce a desired playing time given the size of the medium.
  - Where a lower sound quality is acceptable, or is demanded by limited bandwidth, sub-multiples of 44.1kHz are used: 22.05kHz is commonly used for audio destined for delivery over the Internet, while 11.025kHz is sometimes used for speech.
  - 48kHz is used when the best quality is desired, e.g., DAT (digital audio tape) recorders and the better sound cards.

Digital Data Acquisition

A general-purpose image processing system

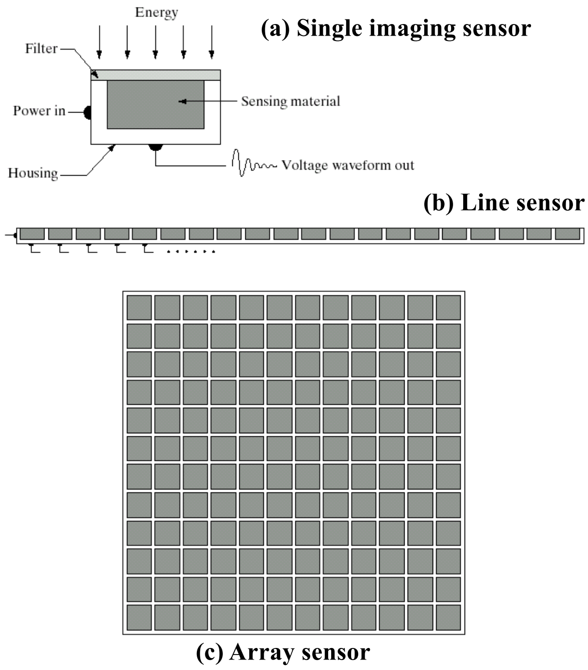


Example: In a digital video camera, the sensors produce an electrical output proportional to light intensity. The digitizer converts these outputs to digital data.

A physical device that is sensitive to the energy radiated by the object we wish to image

A digitizer is a device for converting the output of the physical sensing device into digital form.

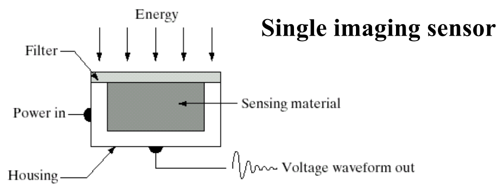
Digital Data Acquisition



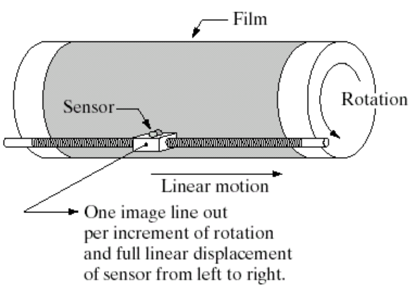
The three principal sensor arrangements used to transform illumination energy into digital images.

Incoming energy is transformed into a voltage by the combination of input electrical power and sensor material that is responsive to the particular type of energy being detected. The output voltage waveform is the response of the sensor(s), and a digital quantity is obtained from each sensor by digitizing its response.

Digital Data Acquisition



In order to generate a 2D image using a single sensor, there has to be relative displacements in both the *x*- and *y*-directions between the sensor and the area to be imaged.

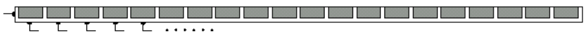


Combining a single sensor with motion to generate a 2D image

Example: an arrangement used in high-precision scanning, where a film negative is mounted onto a drum whose mechanical rotation provides displacement in one dimension. The single sensor is mounted on a lead screw that provides motion in the perpendicular direction. Since mechanical motion can be controlled with high precision, this method is an inexpensive (but slow) way to obtain high-resolution images.

Digital Data Acquisition

Line sensor



A geometry that is used much more frequently than single sensors consists of an in-line arrangement of sensors in the form of a sensor strip.

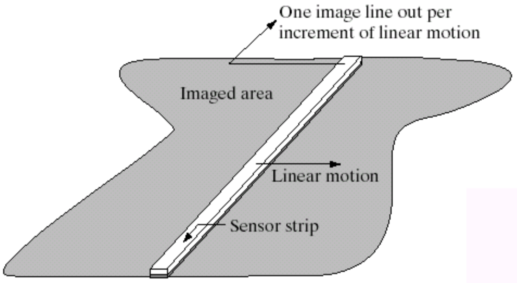


Image acquisition using a linear sensor strip

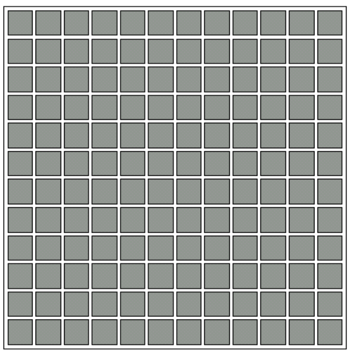
The strip provides imaging elements in one direction. Motion perpendicular to the strip provides imaging in the other direction.

The imaging strip gives one line of an image at a time, and the motion of the strip completes the other dimension of a 2D image.

This is the type of arrangement used in most flat bed scanners.

Digital Data Acquisition

Array sensor

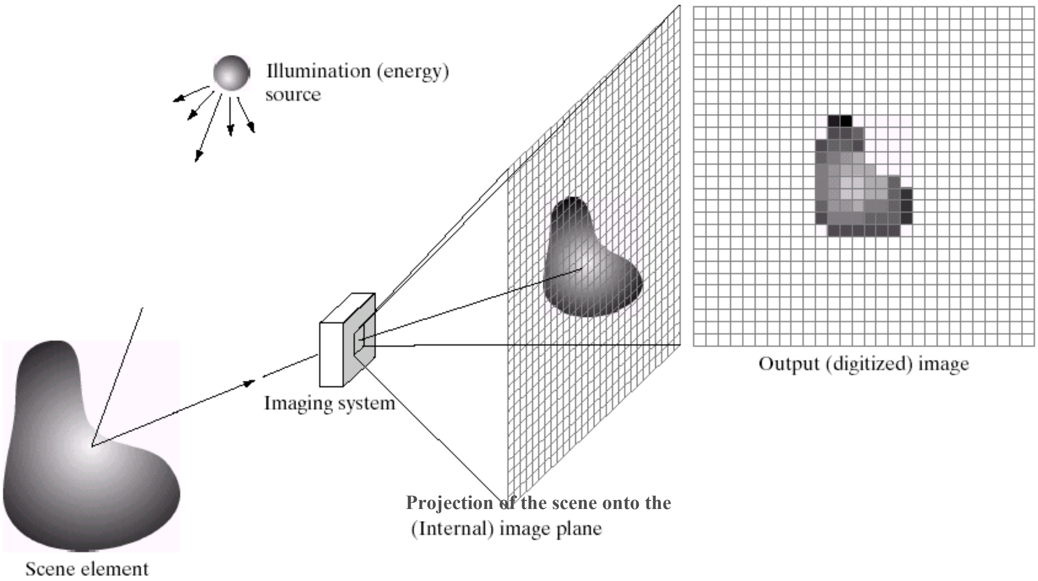


Individual sensors arranged in the form of a 2D array.

Numerous electromagnetic and some ultrasonic sensing devices frequently are arranged in an array format. This is also the predominant arrangement found in digital cameras.

Digital Data Acquisition

An example of the digital image acquisition process



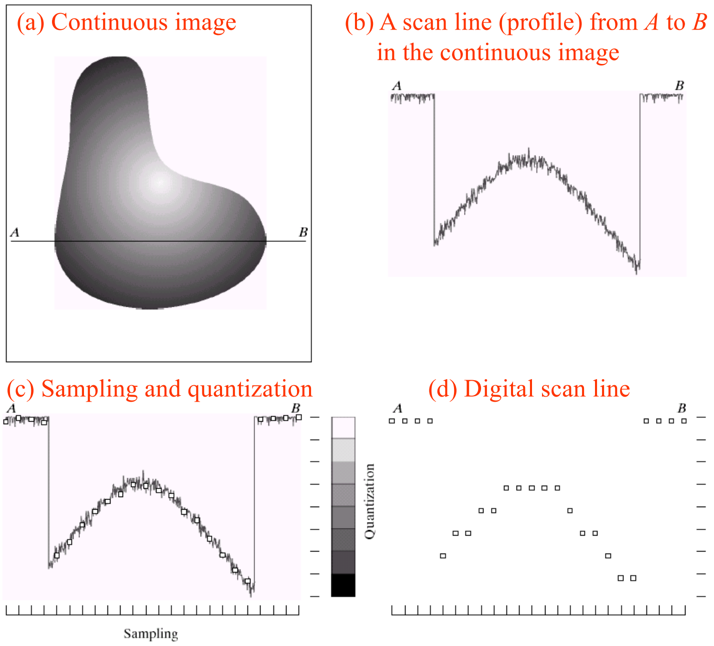
Digital Data Acquisition

- There are numerous ways to acquire images, but our objective in all is the same: to generate digital images from sensed data. The output of most sensors is a continuous voltage phenomenon being sensed.

- To create a digital image, we need to convert the continuous sensed data into digital form. This involves two processes: *sampling* and *quantization*.

Digital Data Acquisition

Image sampling and quantization



(a) Continuous image

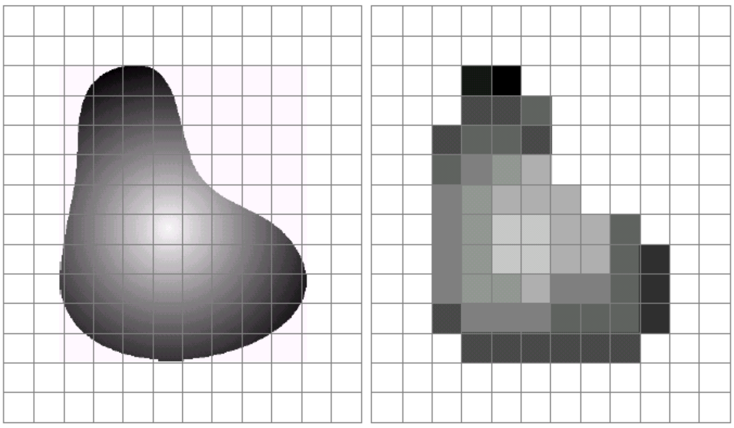
(b) A scan line (profile) from *A* to *B* in the continuous image

(c) Sampling and quantization

(d) Digital scan line

Digital Data Acquisition

Image sampling and quantization

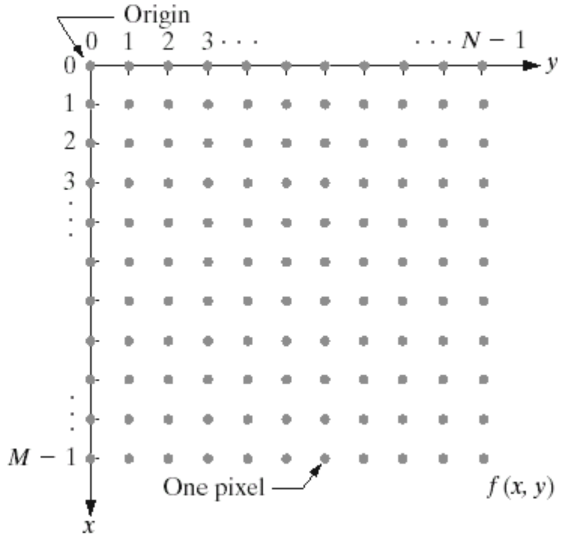


(a) Continuous image projected onto a sensor array

(b) Result of image sampling and quantization

Digital Data Acquisition

Representing digital images



The result of sampling and quantization is a matrix of real numbers.

Visual Perception

- Although the digital image / video processing field is built on a foundation of mathematical and probabilistic formulations, human intuition and analysis play a central role in the choice of one technique versus another, and this choice often is made based on subjective, visual judgments.
- Developing a basic understanding of *human visual perception* as a first step in the study of multimedia is appropriate.

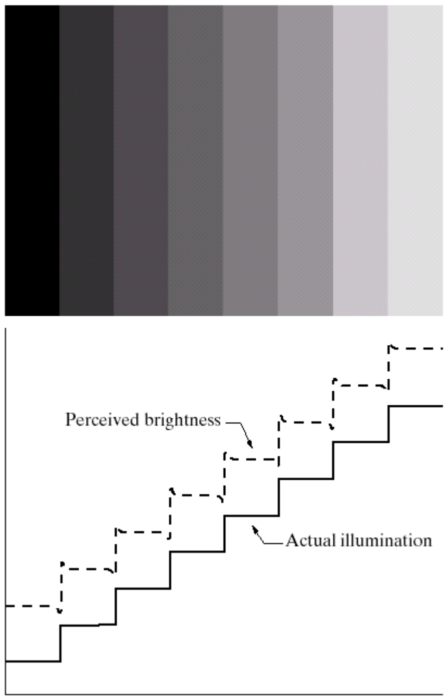
Visual Perception

Brightness adaptation and discrimination

- Because digital images are displayed as a discrete set of intensities, the human eye's ability to discriminate between different intensity levels is an important consideration in presenting image-processing results.
- Experimental evidence indicates that subjective brightness (intensity as perceived by the human visual system) is not a simple function of intensity.

Visual Perception

Phenomenon 1: *Mach bands*



Although the intensity of the stripes is constant, we actually perceive a brightness pattern that is strongly scalloped, especially near the boundaries.
These seemingly scalloped bands are called *Mach bands* after Ernst Mach, who first described the phenomenon in 1865.

This phenomenon clearly demonstrate that perceived brightness is not a simple function of intensity. The human visual system tends to undershoot or overshoot around the boundary of regions of different intensities.

Visual Perception

Phenomenon 2: *simultaneous contrast*

All the center squares have exactly the same intensity. However, they appear to the eye to become progressively darker as the background becomes lighter.

A more familiar example is a piece of paper that seems white when lying on a desk, but can appear totally black when used to shield the eyes while looking directly at a bright sky.

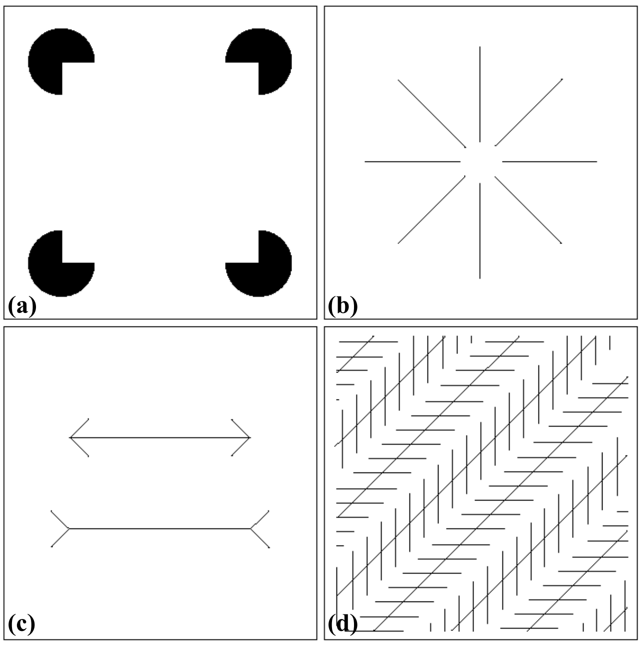
Visual Perception

Phenomenon 3: *optical illusion*

The eye fills in non-existing information or wrongly perceives geometrical properties of objects.

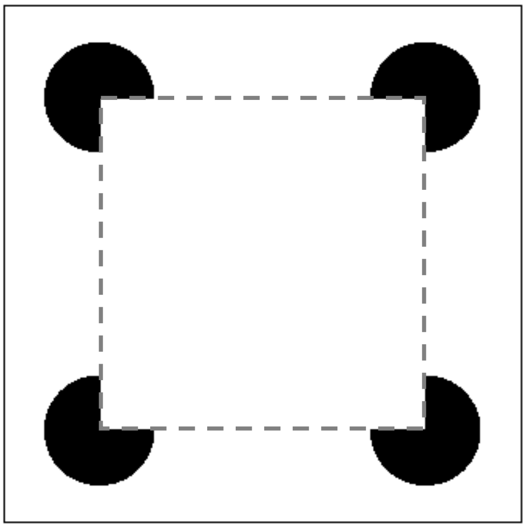
<u>Optical illusion is a characteristic of the human visual system that is not fully understood</u>.

(a) The outline of a square is seen clearly, in spite of the fact that no lines defining such a figure are part of the image;
(b) The same effect, but with a circle. Note how just a few lines are sufficient to give the illusion of a complete circle;
(c) The two horizontal line segments are of the same length, but one appears shorter than the other;
(d) All lines that are oriented at 45˚ are equidistant and parallel. Yet the crosshatching creates the illusion that those lines are far from being parallel.

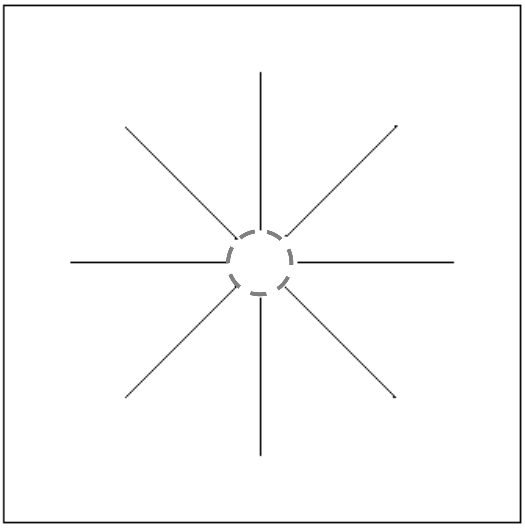


Application of Visual Perception : Edge Detection

The outline of a square is seen clearly, in spite of the fact that no lines defining such a figure are part of the image。



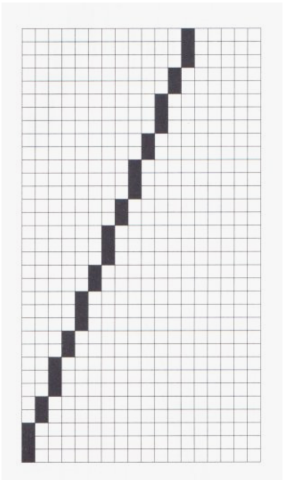
The same effect, but with a circle. Note how just a few lines are sufficient to give the illusion of a complete circle。

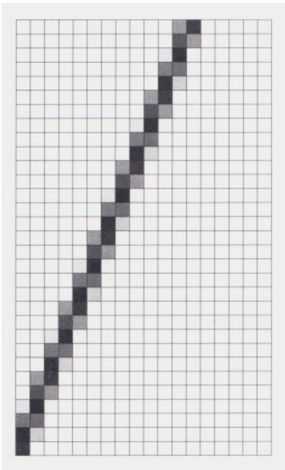


- So strong is this perception that the dotted virtual curves provided in the above images need not be provided to a human at all.
- A machine vision system will not be fooled into perceiving that the central region is brighter than the background because it has objective pixel intensities.

Application of Visual Perception : Anti-aliasing

Approximating abstract shapes on a grid of finite pixels leads to "*jaggies*" or "*staircasing*" effect. *Anti-aliasing* technique can offset this effect.





We can soften the effect by using intermediate grey values for some pixels. We cannot simply tone down the black pixels to produce a grey line instead of a black one; we want to try to use a range of greys to convey to the eye and brain of someone looking at the displayed line the appearance of a smoothness that cannot actually be achieved by the finite pixels.

Application of Visual Perception : Motion Blur

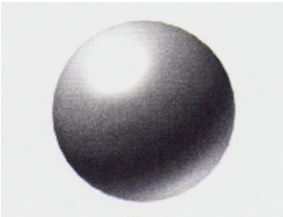
Motion blur in computer animation

- When recording reality with a video or a film camera, we observe that objects that move too fast in front of the camera appear blurred. This phenomenon is called **motion blur**, and it occurs naturally in film or video recordings where the shutter speed is too slow to freeze an object in motion.
- Motion blur can add a touch of realism to computer animation because it reminds viewers of the blurring effect that occurs when we record fast-moving real objects directly with a camera. But motion blur does not occur naturally in computer animation, it must be added.

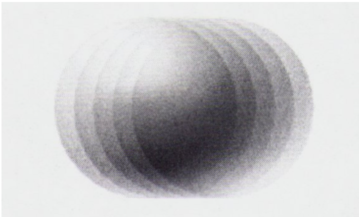
Application of Visual Perception : Motion Blur

One of the common techniques programs use to create motion blur is rendering the scene a number of times while advancing the animation slightly. The multiple images are then composited together into a single, motion-blurred image. However, this means a 4-5 fold increase in rendering time.

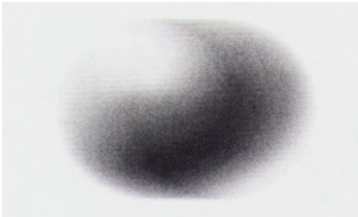
Another way to create the effect is by applying a directional blurring filter in post-production using either a plug-in or paint program. This type of blur is generally more realistic, and renders faster as well.



Unaffected object



5-step motion blur created by 3D rendering



20-pixel Motion Blur applied with Photoshop

Application of Visual Perception : Image Compression

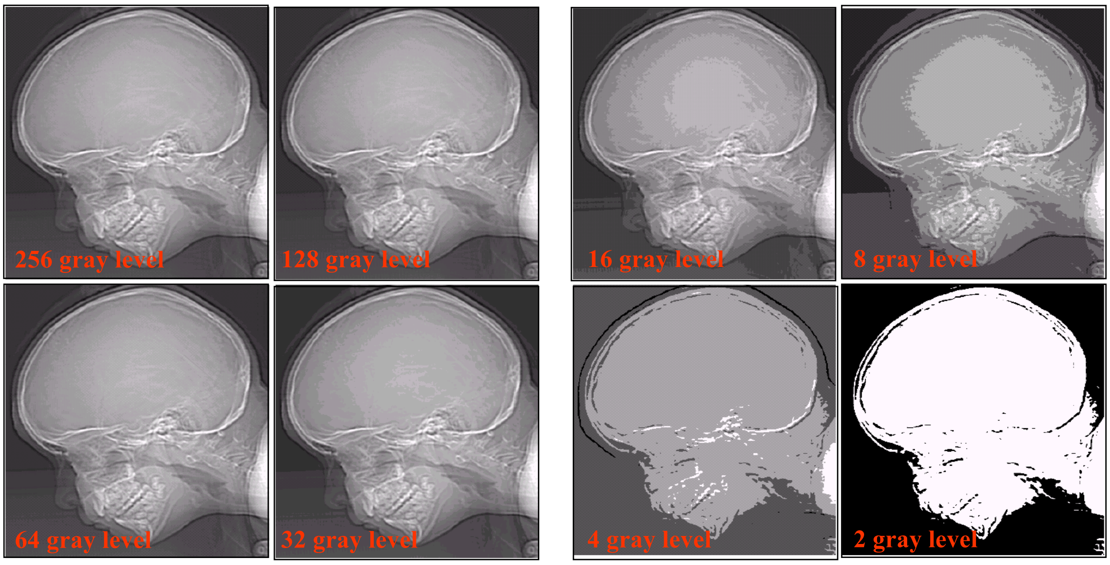
Human eye does not respond with equal sensitivity to all visual information. Certain information simply has less relative importance than other information in normal visual processing. This information is said to be *psycho-visually redundant*. It can be eliminated without significantly impairing the quality of image perception.

Application of Visual Perception : Image Compression

That psycho-visual redundancies exist should not come as a surprise, because human perception of the information in an image normally does not involve quantitative analysis of every pixel value in the image.

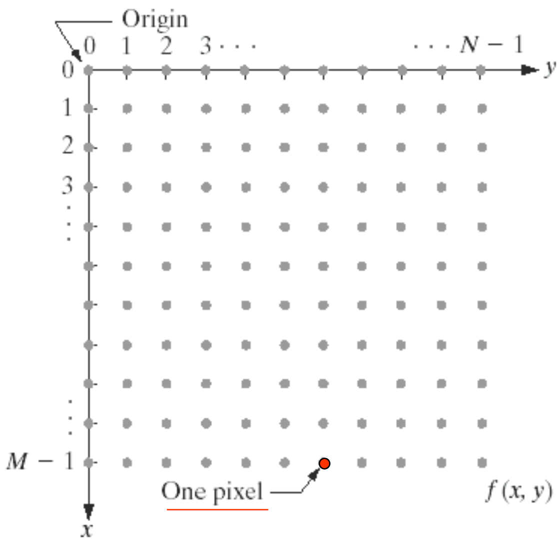
In general, an observer searches for distinguishing features such as edges or textural regions and mentally combines them into recognizable groupings. The brain then correlates these groupings with prior knowledge in order to complete the image interpretation process.

Application of Visual Perception : Image Compression



The 32-level image has an almost imperceptible set of very fine ridge-like structures in areas of smooth gray levels (particularly in the skull). This effect, caused by the use of an insufficient number of gray levels in smooth areas of a digital image, is called *false contouring*, so called because the ridges resemble topographic contours in a map. False contouring generally is quite visible in images displayed using 16 or less uniformly spaced gray levels.

Digital Image Representation



$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix}$$

$$\mathbf{A}=\begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}$$

Each element of the matrix array is called an *image element*, *picture element*, ***pixel***, or *pel*.

Digital Image Representation

- An image is referred to as a 2D light intensity function $f(x,y)$ where
  - (x,y) denotes the spatial coordinate, and
  - f is a function of (x,y) and is proportional to the brightness or grey level of the image at that point
- Geometrically:

Digital Image Representation

- The image digitization process requires decisions about values for ***M*** (rows), ***N*** (columns), and for the number , ***L***, of discrete gray levels allowed for each pixel.

$$L = 2^k$$

  - *Due to processing, storage, and sampling hardware considerations, the number of gray levels typically is an integer power of 2.*

- The number, **b**, of bits required to store a digitized image is

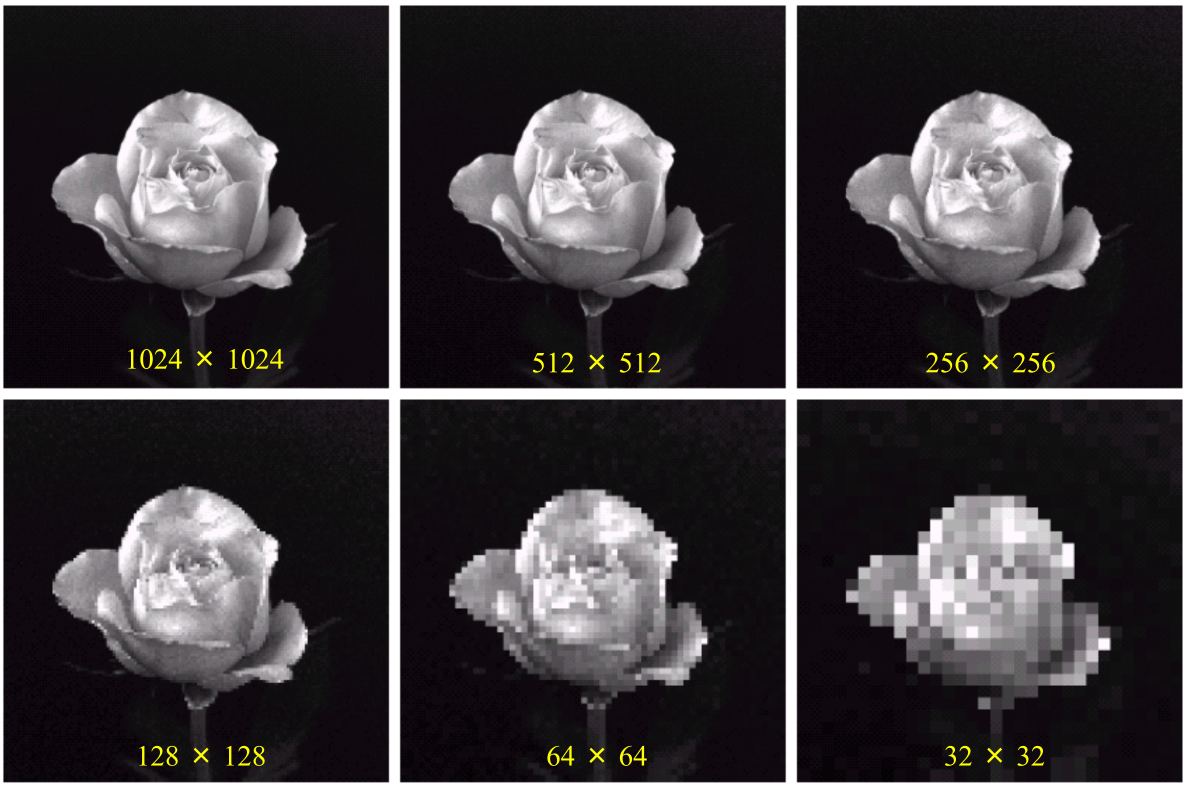
$$b = M \times N \times k$$

- When an image can have 2^k gray levels, it is common practice to refer to the image as a "***k*-bit image**".

Spatial Resolution of Image

- Sampling is the principal factor determining the <u>spatial resolution</u> of an image.
- Basically, spatial resolution is the smallest discernible detail in an image.

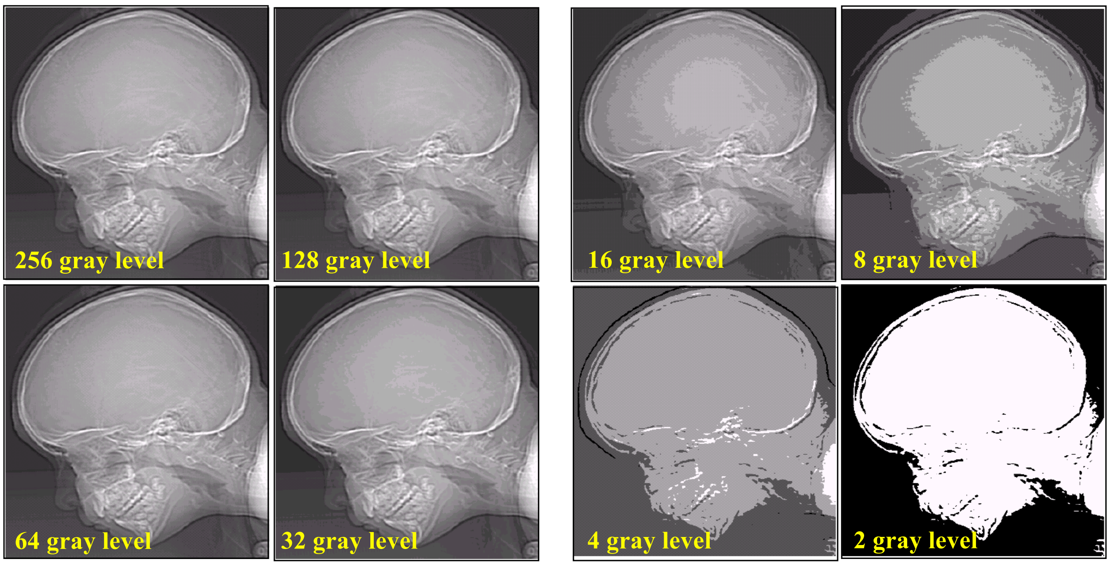
Spatial Resolution of Image



Gray-Level Resolution of Image

- Gray-level resolution refers to the smallest discernible change in gray level.
- The most common number is 8 bits, with 16 bits being used in some applications where enhancement of specific gray-level ranges is necessary.

Gray-Level Resolution of Image



Digital Image Processing

Image processing approaches fall into two broad categories: spatial domain methods and frequency domain methods.

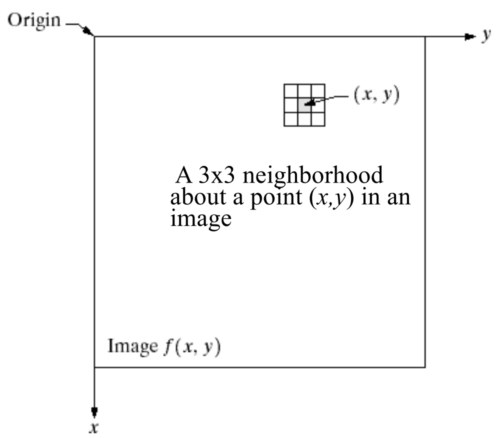
- The spatial domain processing techniques are based on direct manipulation of pixels in an image;
- The frequency domain processing techniques are based on modifying the Fourier (or others) transform of an image.

Spatial Domain Image Processing

Spatial domain processes can be denoted by the expression:

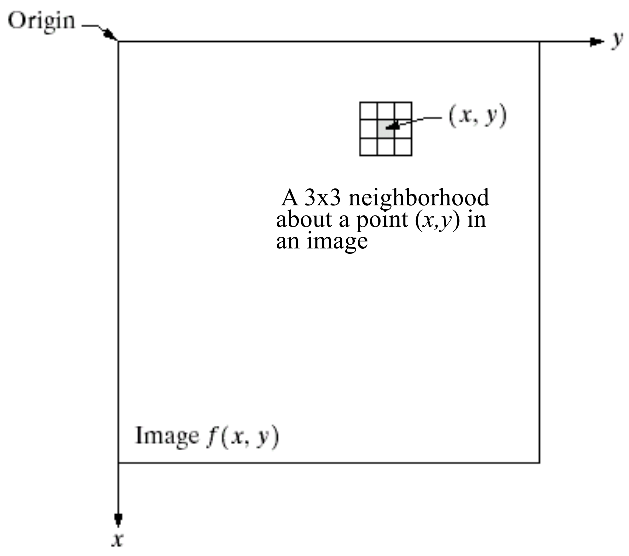
$$g(x,y) = T[f(x,y)]$$

where $f(x,y)$ is the input image, $g(x,y)$ is the processed image, and T is an operator on f, defined over some neighborhood of (x,y).



* Such approach is referred to as **pixel group processing**, **mask processing** or **filtering**.

Spatial Domain Image Processing



The principal approach in defining a neighborhood about a point (*x,y*) is to use a square or rectangular sub-image area centered at (*x,y*). The center of the sub-image is moved from pixel to pixel starting at the top left corner. The operator *T* is applied at each location (*x,y*) to yield the output, *g*, at that location. The process utilizes only the pixels in the area of the image spanned by the neighborhood.
It computes each pixel's new value as a function not just of its old value, but also of the values of neighbouring pixels.

Pixel Point Processing

Spatial domain processes can be denoted by the expression:

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

The simplest form of *T* is when the neighborhood is of size 1 × 1 (a single pixel). In this case, *g* depends only on the value of *f* at (*x,y*), and *T* becomes a *gray-level* (also called an *intensity* or *mapping*) *transformation function* of the form

$$s = T\,(\,r\,)$$

where *r* and *s* are variables denoting, respectively, the gray level of *f*(*x,y*) and *g*(*x,y*) at any point (*x,y*).

It computes a pixel's new value solely on the basis of its old value, without regard to any other pixel.

* Such approach is referred to as **pixel point processing**.

Spatial Domain Image Processing

Linear and Nonlinear Operations

- Let T be an operator whose input and output are images. T is said to be a ***linear*** operator if, for any two images f and g and any two scalars a and b,

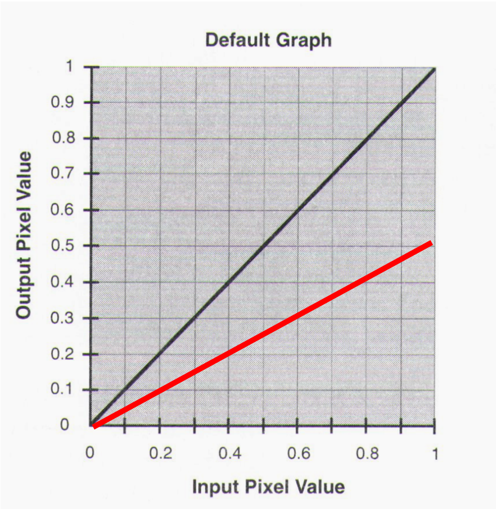
$$T(af + bg) = aT(f) + bT(g)$$

- The result of applying a linear operator to the sum of two images is identical to applying the operator to the images individually, multiplying the results by the appropriate constants, and then adding those results.
- An operator that fails the test of the equation is by definition ***nonlinear***.
- Linear operations are exceptionally important in image processing because they are based on a significant body of well-understood theoretical and practical results. Although nonlinear operations sometimes offer better performance, they are not always predictable, and for the most part are not well understood theoretically.

Image Enhancement

- Image enhancement is one of the most interesting and visually appealing areas of image processing.
- To process an image so that the result is suitable than the original image for a specific application.
- Image features such as boundaries, edges, and contrast, etc, are enhanced for display and analysis.
- The information of the image will not be increased but the chosen features
- Examples of enhancement
  - contrast stretching
  - histogram equalization
  - noise smoothing
  - filtering

Function / Parameter Curves



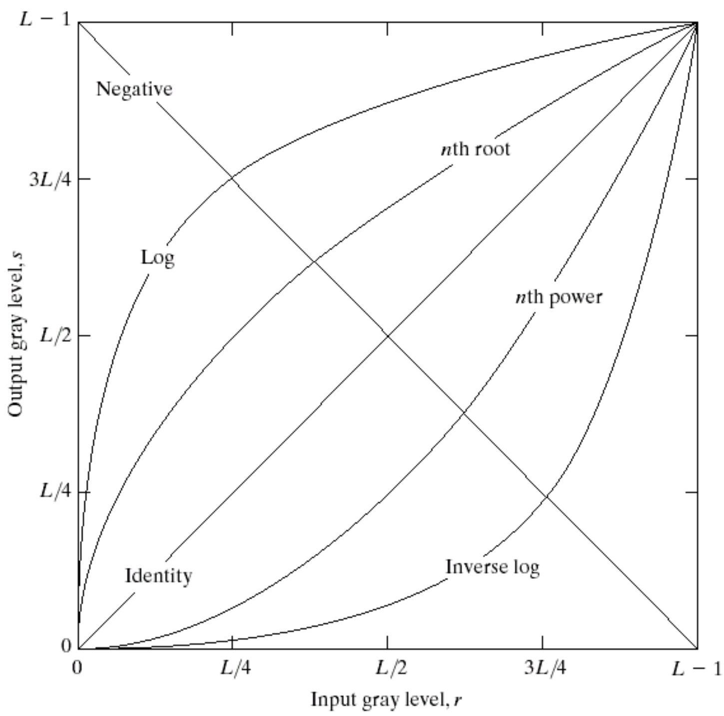
Graph of an unmodified image

The **function or parameter curves** are graphs that represent and control different attributes of an image, such as brightness or color. These attributes can be easily modified by manipulating the function curves without having to alter the image directly with a retouching tool. Making image manipulations that involve all of the image or large portions of it are best performed with function curves.

Function curves for image manipulation are usually represented by a line that starts at the lower left corner of a square and ends at the upper right corner. The straight diagonal line represents one or several untouched attributes of the original image. Any changes made to the line will result in changes to the image.

Image Enhancement

- Gray Level Transformations (GLT)



$$s = T(r)$$

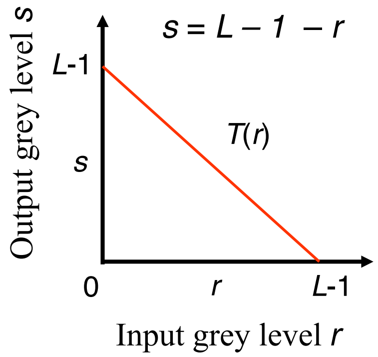
Pixel Point Processing

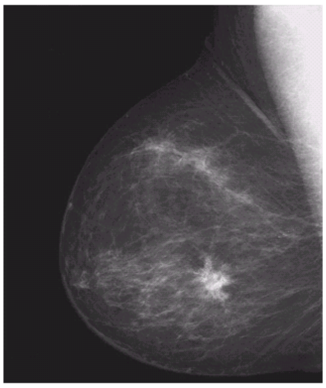
Three basic types of GLT functions used for image enhancement:

- **Linear** (negative and identity);
- **Logarithmic** (log and inverse-log);
- **Power-law** (*n*th power and *n*-th root)

Linear Transformation – Image negatives

- useful in displaying medical images and film processing, etc
- particularly suited for enhancing white or gray detail embedded in dark regions, especially when the black areas are dominant in size.
- reverse the order of pixel intensities





Original Mammogram Image

Negative Image

Log Transformation

Dynamic range compression

- sometimes the dynamic range of an image exceeds the capability of the display device: only the brightest parts of the image are visible on the display.
- can map a narrow range of low gray-level values in the input image into a wider range of output levels. The opposite is true of higher values of input levels.
- allow to expand the values of dark pixels in an image while compressing the higher-level values.

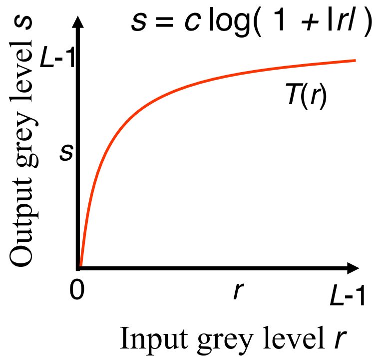
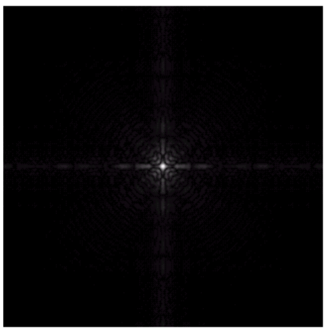
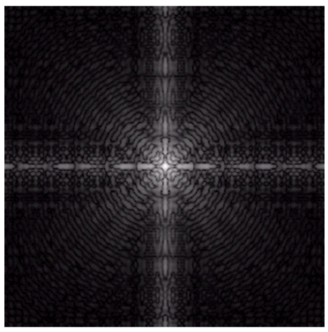


Image with large dynamic range (pixel values: 0 ~ 1.5 x 10^6)



Original image of Fourier spectrum

The range of values of the result become 0 to 6.2 (with c = 1 scaling)



Result of log transformation.

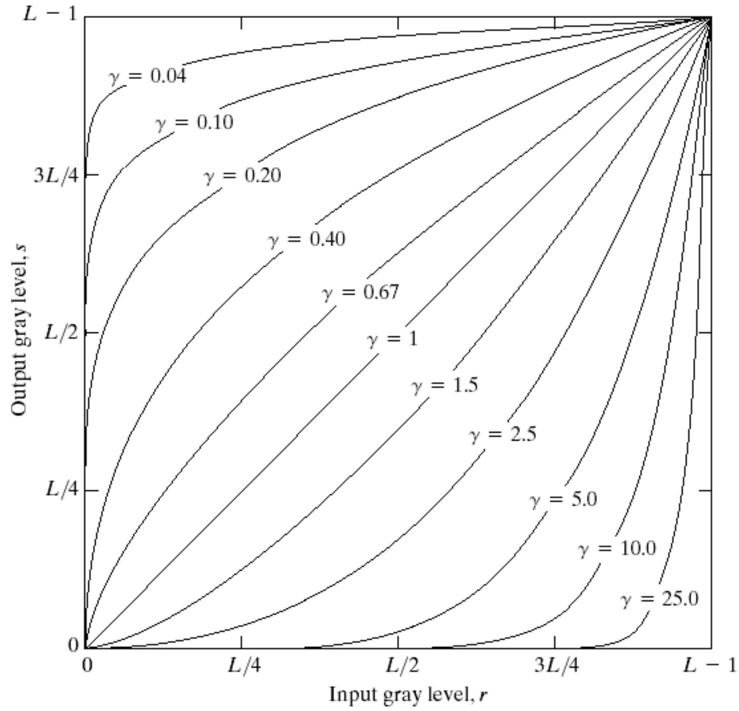
Power-Law Transformation

$$s = c\,r^{\gamma}$$

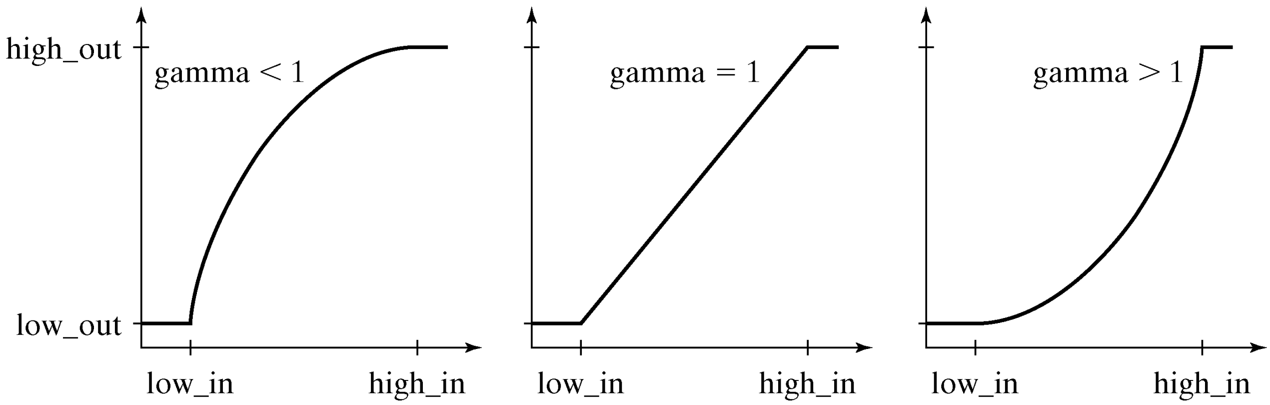
where c and γ (*gamma*) are positive constants.

- Power-law curves with fractional values of γ map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher values of input levels.
- Unlike the log function, a family of possible transformation curves obtained simply by varying γ.

Plots of $s = cr^{\gamma}$ for various values of γ (c = 1 in all cases)



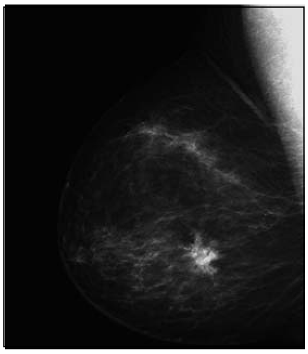
Power-Law Transformation



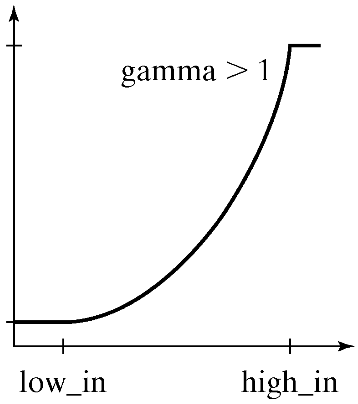
Power-Law Transformation

Example - compressing the low end and expanding the high end of the gray scale.

Original
Mammogram Image

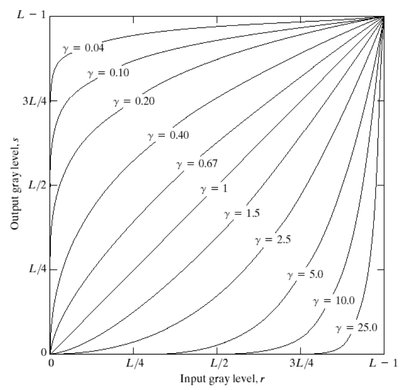


Result of enhancing
the image with
gamma=2



Power-Law Transformation

- Contrast manipulation



MR image of
a fractured human spine

The original image is
predominantly dark

$\gamma = 0.6$

$\gamma = 0.4$

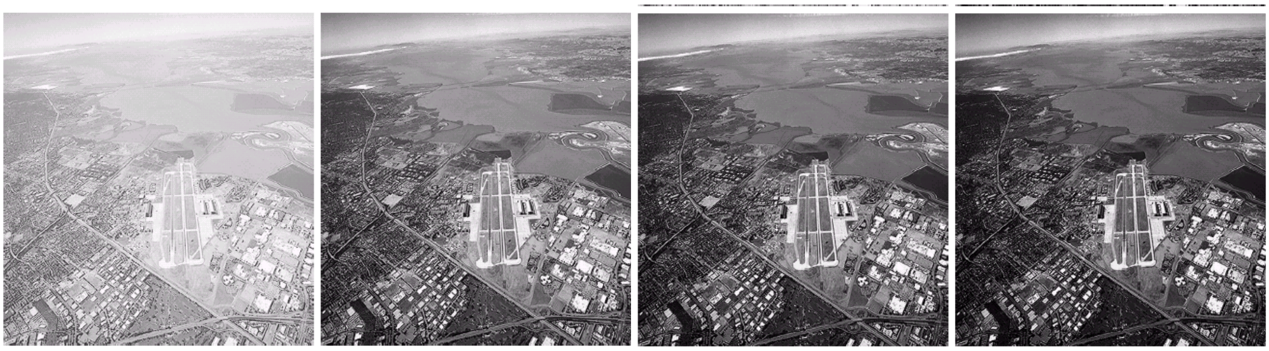
$\gamma = 0.3$

NOTE: As gamma decreased from 0.6 to 0.4, more detail became visible. A further decrease of gamma to 0.3 enhanced a little more detail in the background, but began to reduce contrast to the point where the image started to have a very slight "washed-out" look, especially in the background.

Power-Law Transformation

- Contrast manipulation

Aerial image from NASA



Original image | $\gamma = 3.0$ | $\gamma = 4.0$ | $\gamma = 5.0$

NOTE: The image to be enhanced here has a washed-out appearance, indicating that a compression of gray levels is desirable. Suitable results were obtained with gamma values of 3.0 and 4.0, the latter having a slightly more appealing appearance because it has higher contrast. The result obtained with $\gamma = 5.0$ has areas that are too dark, in which some detail is lost.

Piecewise-Linear Transformation

- A complementary approach to the previous three basic transformations is to use piecewise linear functions.

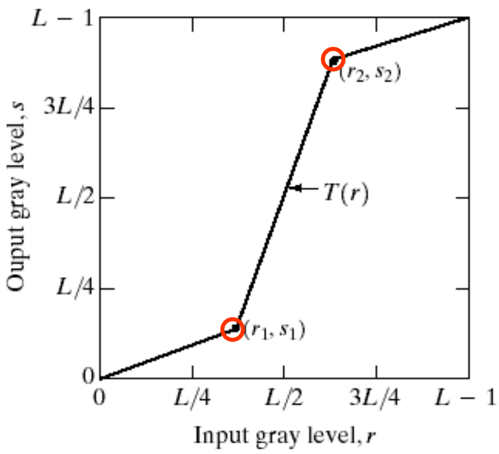
- Advantage: It can be arbitrarily complex.
- Disadvantage: It requires considerably more user input.

- Two typical functions: *contrast-stretching* and *gray-level slicing*.

Piecewise-Linear Transformation

– Contrast stretching

A typical PLT function



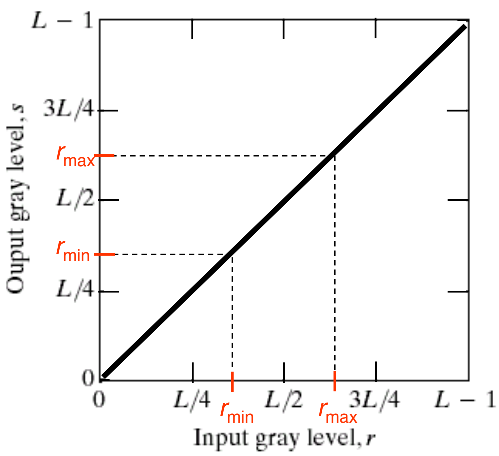
The locations of points (r_1, s_1) and (r_2, s_2) **control** the shape of the transformation function.

- Low-contrast images can result from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture during image acquisition.
- The idea behind contrast stretching is to <u>increase the dynamic range of the gray levels</u> in the image being processed.

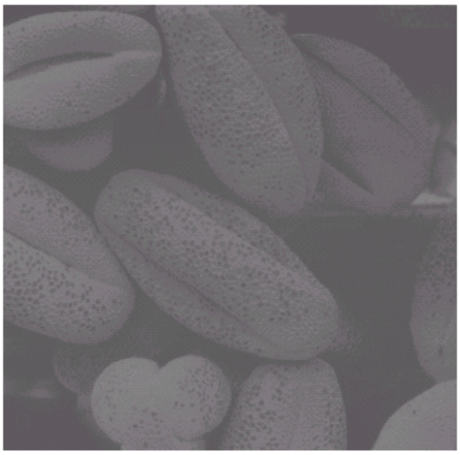
Piecewise-Linear Transformation

– Contrast stretching

A PLT function (Identity)



NOTE: r_{min} and r_{max}, denote the minimum and maximum gray levels in the original image, respectively.

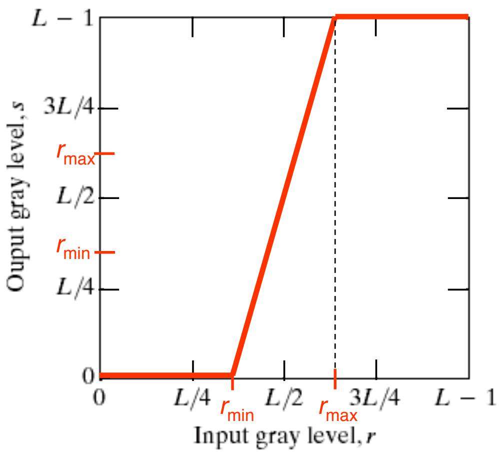


Original image with low contrast

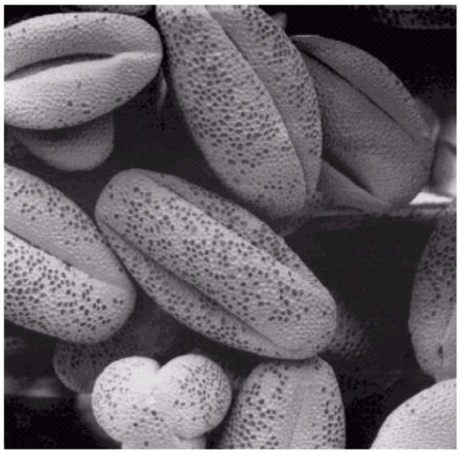
Piecewise-Linear Transformation

– Contrast stretching

A PLT function (contrast stretching)



NOTE: r_{min} and r_{max}, denote the minimum and maximum gray levels in the original image, respectively.

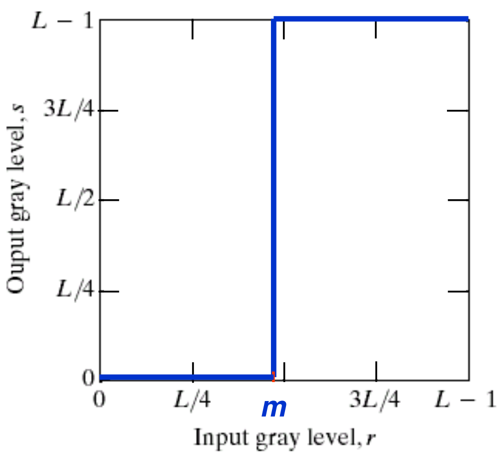


Result of contrast stretching, obtained by setting:
$(r_1, s_1) = (r_{min}, 0)$
$(r_2, s_2) = (r_{max}, L\text{-}1)$
(shown as the red line)

Piecewise-Linear Transformation

– Contrast stretching

A PLT function (contrast stretching)



NOTE: m, denotes the mean gray level in the original image.

Result of contrast stretching, obtained by setting:
$r_1 = r_2$
$s_1 = 0$; $s_2 = L\text{-}1$
(shown as the blue line)

* The transformation becomes a **thresholding function** that creates a binary image.

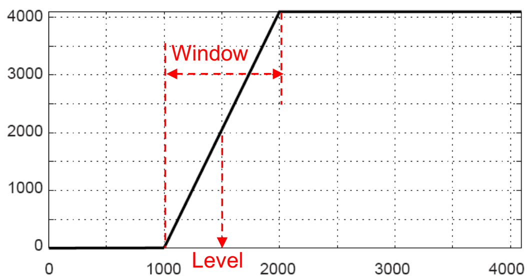
Window-Level Operation (a.k.a. Window-Center Adjustment)

- A particular and popular GLT is the Window-Level operation (also know as Window-Center Adjustment). In this operation, an interval or **window** is selected in the original gray level range, determined by the window center or level l, and the window width w. Explicitly

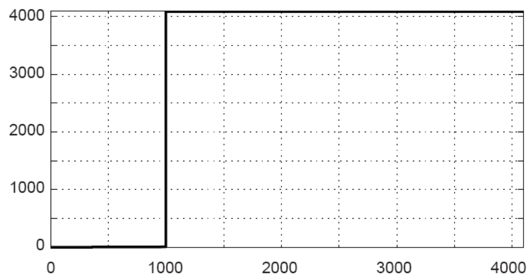
$$T_{l,w}(r)=\begin{cases} 0, & \text{for } r < l-\frac{w}{2} \\ \frac{M}{w}\left(r-l+\frac{w}{2}\right), & \text{for } l-\frac{w}{2} \le r \le l+\frac{w}{2} \\ M, & \text{for } r > l+\frac{w}{2} \end{cases}$$

where M is the largest gray level in the original gray level range (dynamic range).

- Contrast outside the window is lost completely, whereas the portion of the range lying inside the window is stretched to the original gray level range.

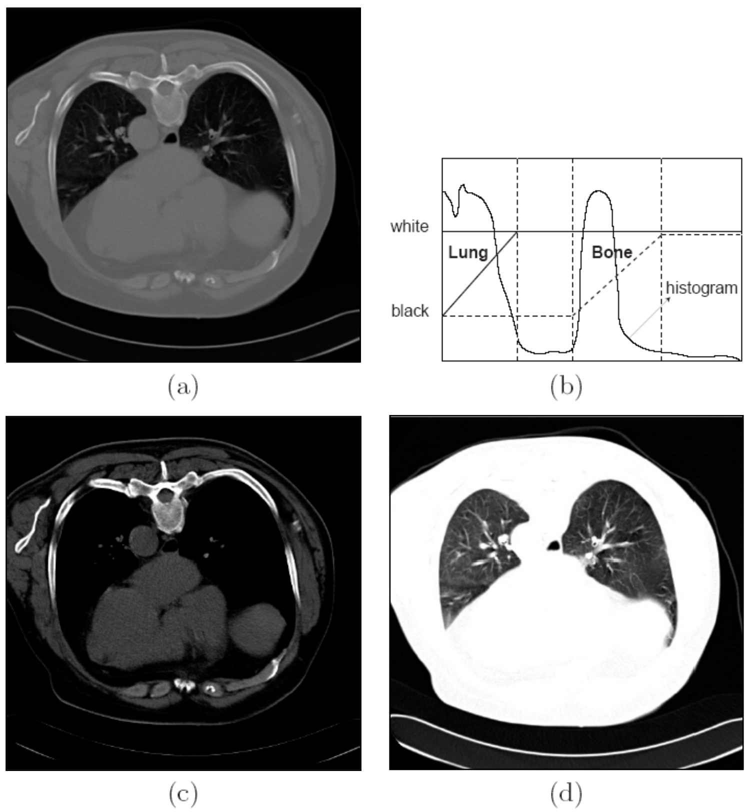


Level / Windowing with l=1500, w=1000



Thresholding with l=1500, w=0

Window-Level Operation

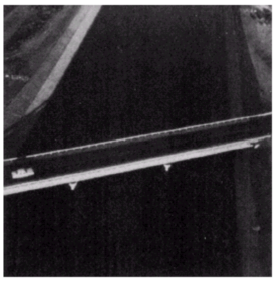


- Original CT image (a) with bimodal histogram (b).
- (c – d) Result of window-level operation using a bone window (dashed line in (b)) and lung window (solid line in (b)), respectively.

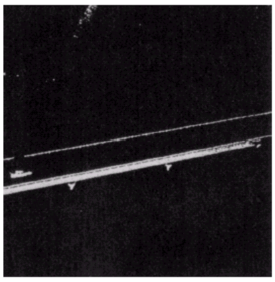
Piecewise-Linear Transformation

– Gray-level slicing

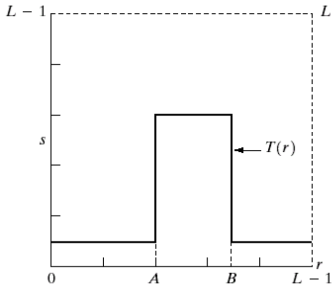
- To highlight a specific range of gray levels in an image, such as enhancing flaws in X-ray images.



Original image

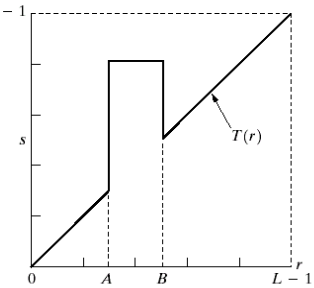


Result of using Approach 1



Approach 1

Highlight range [A, B] of gray levels and reduce all others to a constant level



Approach 2

Highlight range [A, B] but preserve all other levels

Image Enhancement – Histogram Processing

What is Histogram ?

- The *histogram* of a digital image with gray levels in the range [0, L-1] is a discrete function

$$\boxed{h(r_k) = n_k,}$$

where r_k is the k-th gray level and n_k is the number of pixels in the image having gray level r_k.

- A histogram provides a view of the intensity profile of an image and is often displayed as a bar chart.
- Pixel values are partitioned and counted with the population of each partition value placed in its own bin.
- The pixel intensities are plotted along the horizontal x-axis while the number of occurrences for each intensity are plotted along the vertical y-axis.
- Histograms can be viewed as probability density functions.

Image Enhancement – Histogram Processing

What is Histogram ?

Original
Mammogram Image

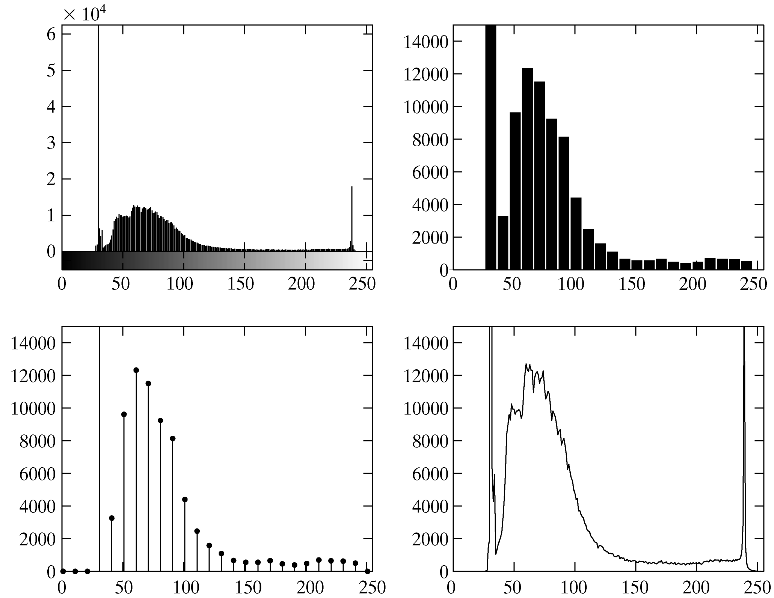


Image Enhancement – Histogram Processing

Normalized Histogram

- It is common practice to normalize a histogram by dividing each of its values by the total number of pixels in the image, denoted by *n*.
- Thus, a *normalized histogram* is given by: $\boxed{p(r_k) = n_k / n}$

  where $p(r_k)$ gives an estimate of the probability of occurrence of gray level r_k.

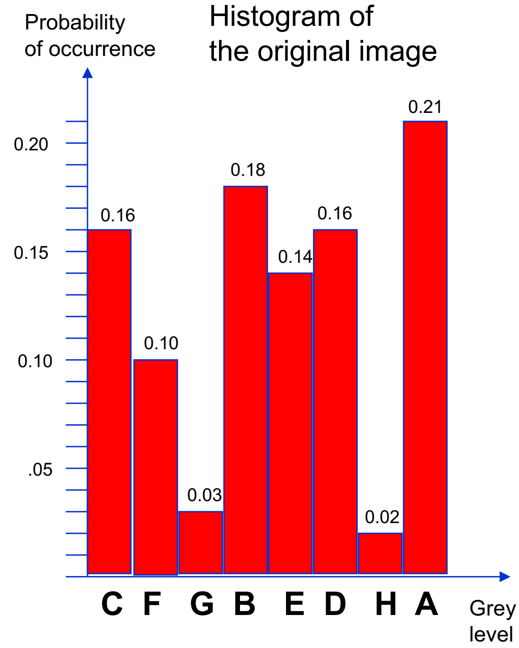
Example_

Image Enhancement – Histogram Processing

Normalized Histogram

A 10 x 10 image with
eight grey levels (C,F,G,B,E,D,H,A)

A	E	F	C	E	C	C	E	F	C
C	C	A	D	A	D	A	D	A	G
B	B	A	C	A	B	A	E	A	E
G	A	F	C	C	F	A	B	B	B
D	D	D	D	H	H	D	E	E	B
D	D	D	D	D	D	D	B	A	B
E	A	E	A	E	B	D	B	B	B
F	C	C	C	F	A	A	B	B	B
E	A	A	E	B	E	E	A	G	A
F	F	B	C	F	C	F	C	C	A



Example_

Four basic image types and their corresponding histograms

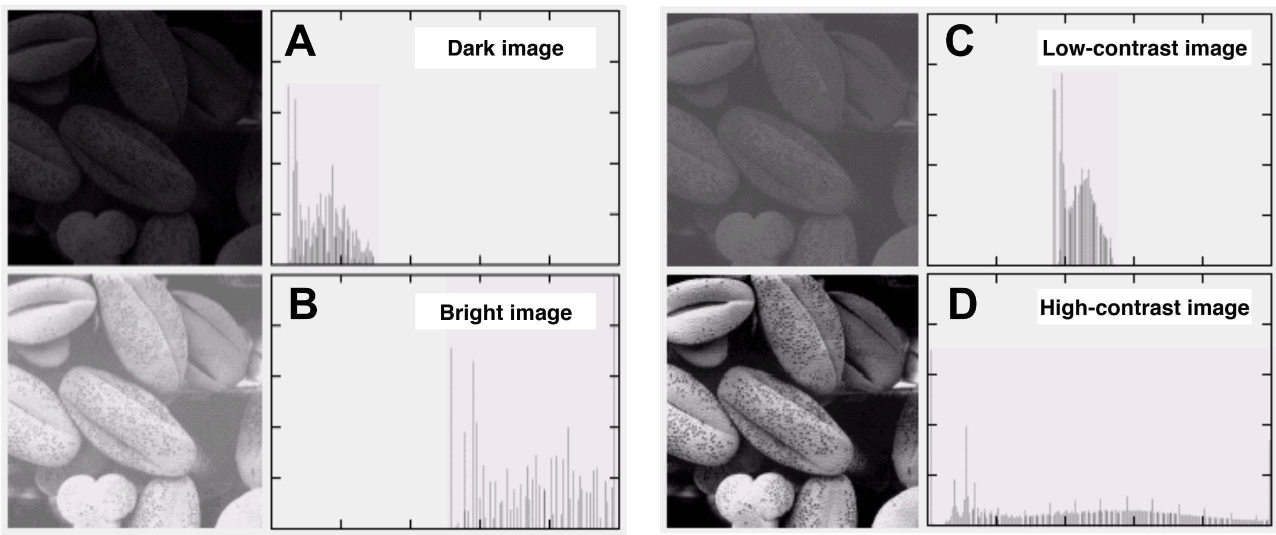


Image Enhancement – Histogram Processing

- An image whose pixels tend to occupy the entire range of possible gray levels and tend to be distributed uniformly, will have an appearance of high contrast and will exhibit a large variety of gray tones. (Figure-D in the previous slide)
- It is possible to develop a transformation function that can automatically achieve the above effect, based only on information available in the histogram of the input image.
- The appearance of the histogram of an image gives useful information for possible contrast enhancement.

Image Enhancement – Histogram Processing

– Histogram equalization

- A mapping to increase the contrast in an image by stretching its histogram to approximately uniformly distributed
- The image that has been histogram equalized always has pixels that reach the brightest grey level

Image Enhancement – Histogram Processing

– Histogram equalization

- *Histogram equalization* requires a mapping (via a transformation) to stretch the histogram of the input image

$$s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n}$$

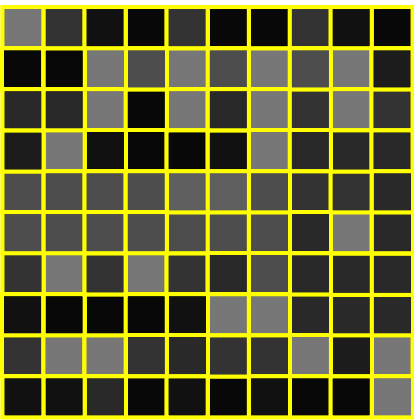
  – r_k is the k-th ($k = 0, 1, \ldots, L\text{-}1$) grey level
  – n_k is the number of pixels in the image with that grey level
  – n is the number of pixels in the image
  – s has as many elements as the original histogram

- $s \in [0,1]$ and it has to be scaled as ss for constructing the resultant image, i.e., $ss \in$ [0, Max-grey-level-value]
- Thus, a processed (output) image is obtained by mapping each pixel with level r_k in the input image into a corresponding pixel with level s_k in the output image.

Question:

Show step by step how to construct the histogram-equalized image from the following original image.

A 10 x 10 original image with
16 grey levels (4-bit image)



8	5	2	1	5	1	1	5	2	1
1	1	8	6	8	6	8	6	8	3
4	4	8	1	8	4	8	5	8	5
3	8	2	1	1	2	8	4	4	4
6	6	6	6	7	7	6	5	5	4
6	6	6	6	6	6	6	4	8	4
5	8	5	8	5	4	6	4	4	4
2	1	1	1	2	8	8	4	4	4
5	8	8	5	4	5	5	8	3	8
2	2	4	1	2	1	2	1	1	8

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Question:

Show step by step how to construct the histogram-equalized image from the following original image.

Example_

Results of histogram equalization

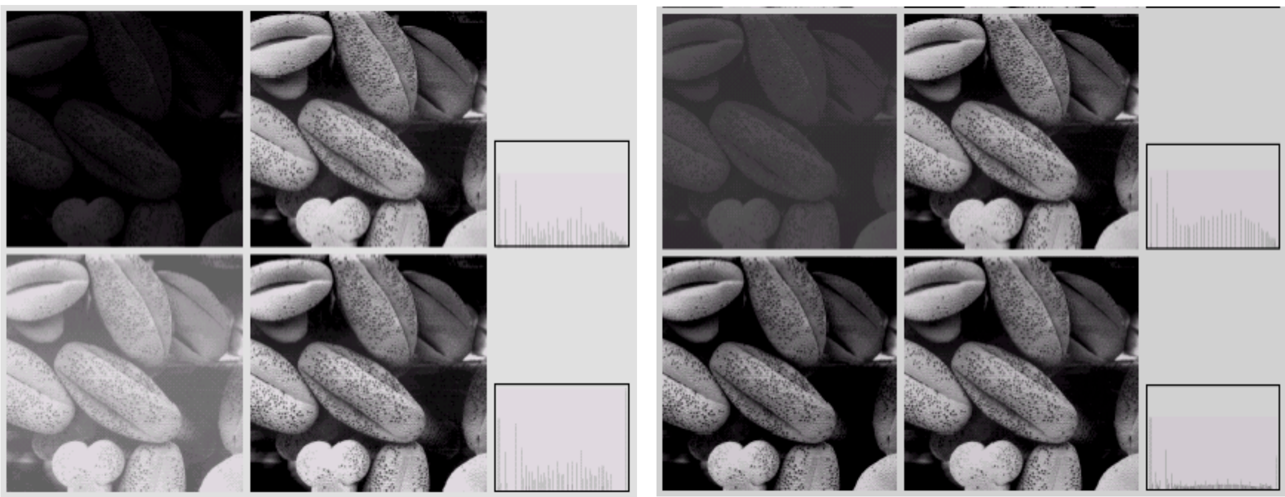
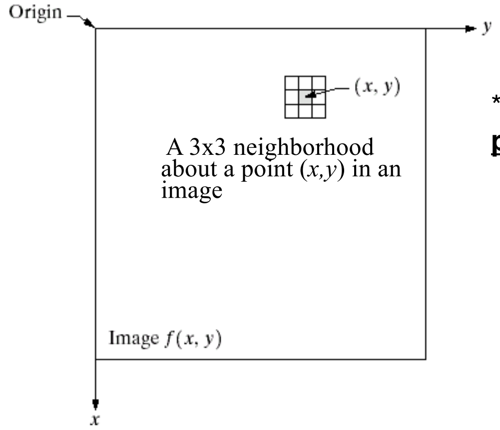


Image Enhancement – Spatial Filtering

Spatial domain processes can be denoted by the expression:

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

where $f\,(x,y)$ is the input image, $g\,(x,y)$ is the processed image, and T is an operator on f, defined over some neighborhood of (x,y).



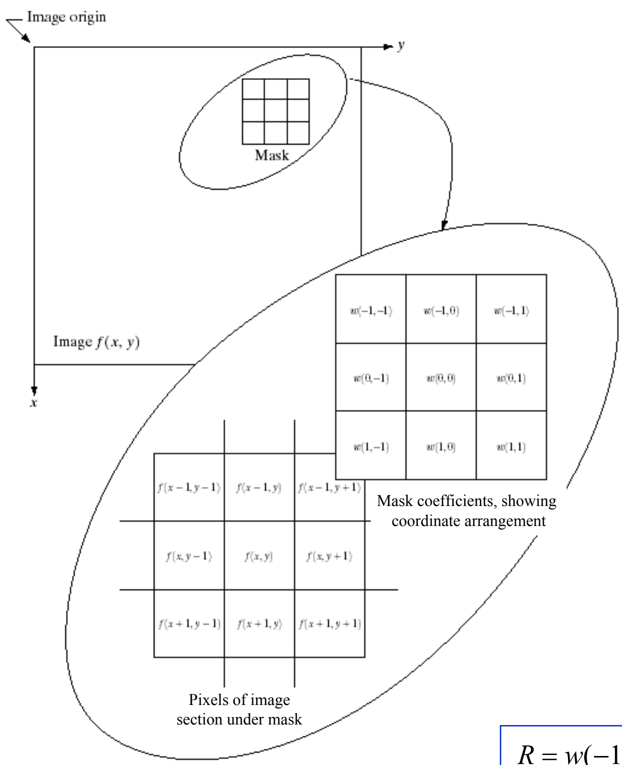
* Such approach is referred to as **pixel group processing**, **mask processing** or **filtering**.

- The sub-image is called a ***(spatial) filter***, ***mask***, ***kernel***, ***template***, or ***window***.
- The values in a filter sub-image are referred to as ***coefficients***, rather than pixels.

Image Enhancement

Spatial Filtering –

- The concept of ***filtering*** has its roots in the use of the Fourier transform for signal processing in the frequency domain. Here we are interested in filtering operations that are performed directly on the pixels of an image.
- We use the term ***spatial filtering*** to differentiate this type of process from the more traditional frequency domain filtering.



The mechanics of spatial filtering

Image Enhancement

- Spatial Filtering

The process consists simply of moving the filter mask from point to point in an image. At each point (x,y), the response of the filter at that point is calculated using a predefined relationship.

For linear spatial filtering, the response is given by a sum of products of the filter coefficients and the corresponding image pixels in the area spanned by the filter mask.

Example: the result (or response), R, of linear filtering with the 3 × 3 filter mask at a point (x,y) in the image is

$$\boxed{\begin{aligned} R = {} & w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \cdots \\ & + w(0,0)f(x,y) + \cdots + w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1) \end{aligned}}$$

Image Enhancement

Spatial Filtering –

- For a mask of size $m \times n$, we assume that $m = 2a + 1$ and $n = 2b + 1$, where a and b are non-negative integers. (mask with odd size)
- In general, linear filtering of an image f of size $M \times N$ with a filter mask of size $m \times n$ is given by the expression:

$$\boxed{g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)}$$

where $a = (m - 1) / 2$ and $b = (n - 1) / 2$.

- To generate a complete filtered image, the above equation must be applied for $x = 0, 1, 2, \ldots, M - 1$ and $y = 0, 1, 2, \ldots, N - 1$.

- The above process of linear filtering is similar to a frequency domain concept called *convolution*, and is often referred to as “convolving a mask with an image”. Similarly, filter masks are sometimes called *convolution masks*.

Image Enhancement

Spatial Filtering – Image smoothing

- Smoothing filters are used for blurring and for noise reduction. Blurring can be used in preprocessing steps, such as removal of small details from an image prior to (large) object extraction, and bridging of small gaps in lines or curves.

- Linear smoothing filters (e.g. *box filter, weighted filter*)
- Non-linear smoothing filters (e.g. *median filter, max filter, min filter*)

Image Enhancement

Linear Smoothing Filters – Averaging filters

- The output of a smoothing, linear spatial filter is simply the average of the pixels contained in the neighborhood of the filter mask. These filters sometimes are called *<u>averaging filters</u>*.

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Box filter

A spatial averaging filter in which all coefficients are equal

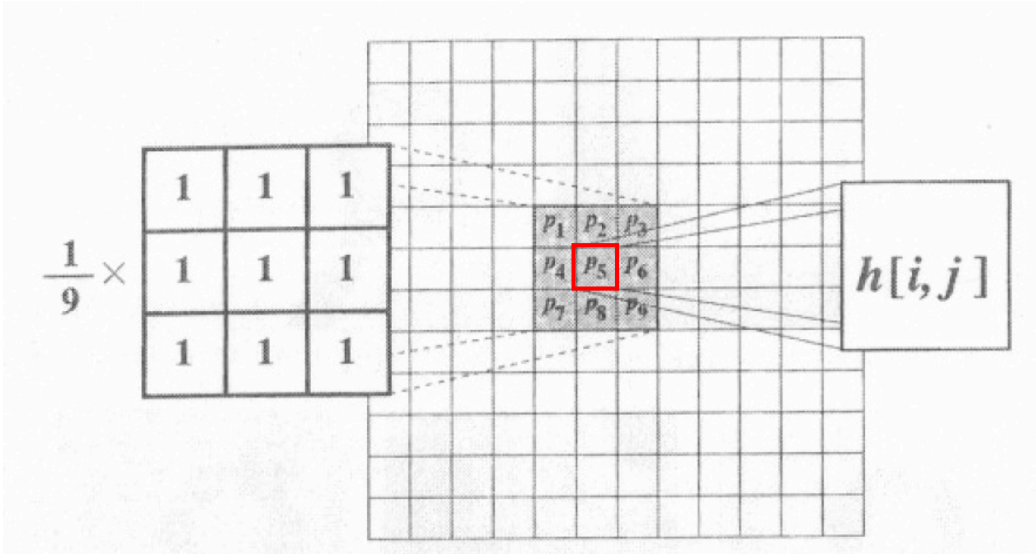
$$\frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Weighted average filter

Pixels are multiplied by different coefficients, i.e. giving more importance (weight) to some pixels at the expense of others.

Linear Smoothing Filters – Averaging filters

3x3 Box Filter



$$h[i,j] = (p_1 + p_2 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9) \times \frac{1}{9}$$

Linear Smoothing Filters – Averaging filters

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	90	0	90	90	90	0	0
0	0	0	90	90	90	90	90	0	0
0	0	0	0	0	0	0	0	0	0
0	0	90	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

3x3 Box Filter →

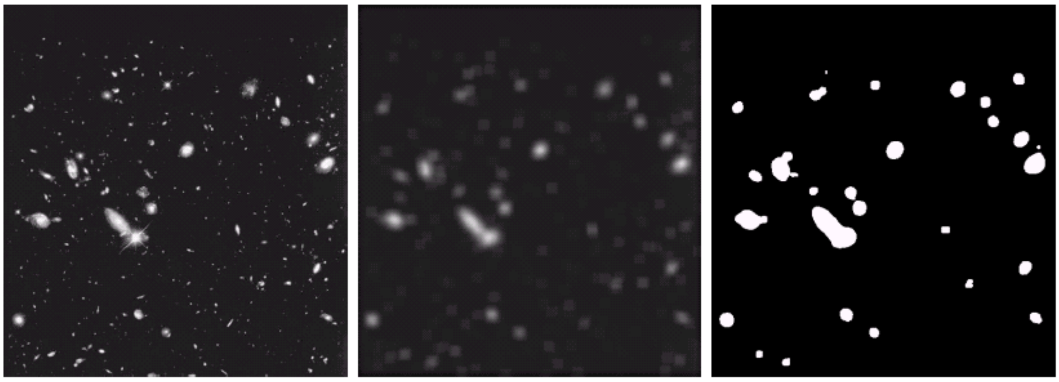
	0	10	20	30	30	30	20	10	
	0	20	40	60	60	60	40	20	
	0	30	60	90	90	90	60	30	
	0	30	50	80	80	90	60	30	
	0	30	50	80	80	90	60	30	
	0	20	30	50	50	60	40	20	
	10	20	30	30	30	30	20	10	
	10	10	10	0	0	0	0	0	

Image Enhancement

Linear Smoothing Filters – Averaging filters

- An important application of spatial averaging is to blur an image for the purpose getting a gross representation of <u>objects of interest</u>, such that the intensity of smaller object blends with the background and larger objects become "blob-like" and easy to detect.

Image from the NASA Hubble Space Telescope



Original image | Smoothing result by a 15 x 15 averaging mask | Result of thresholding

Image Enhancement

Non-linear Smoothing Filters – Order-statistics filters

- <u>Order-statistics smoothing filters</u> are non-linear spatial filters whose response is based on ordering (ranking) the pixels contained in the image area encompassed by the filter, and then replacing the value of the center pixel with the value determined by the ranking result.

- The best-known example is the ***<u>median filter</u>***, which replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel (the original value of the pixel is included in the computation of the median).

Image Enhancement

Non-linear Smoothing Filters – *median filter*

- The median, ξ, of a set of values is such that half the values in the set are less than or equal to ξ, and half are greater than or equal to ξ.
- In order to perform median filtering at a point in an image, we first <u>sort the values of the pixel and its neighbors</u>, <u>determine their median</u>, and <u>assign this value to that pixel</u>.
  - When several values in a neighborhood are the same, all equal values are grouped. For example, suppose that a 3 x 3 neighborhood has values (10,20,20,20,15,20,20,25,100). These values are sorted as (10,15,20,20,20,20,20,25,100), which results in a median of 20.
- The principal function of median filters is to force points with distinct gray levels to be more like their neighbors.
  - In fact, isolated clusters of pixels that are light or dark with respect to their neighbors, and whose area is less than $n^2 / 2$ (one-half the filter area), are eliminated by an n x n median filter. In this case "eliminated" means forced to the median intensity of the neighbors. Larger clusters are affected considerably less.

Non-linear Smoothing Filters – *median filter*

Step-1: sort the pixels into ascending order by their gray level values
Step-2: select the value of the middle pixel as the new value for pixel [i, j]

3x3 Median Filter

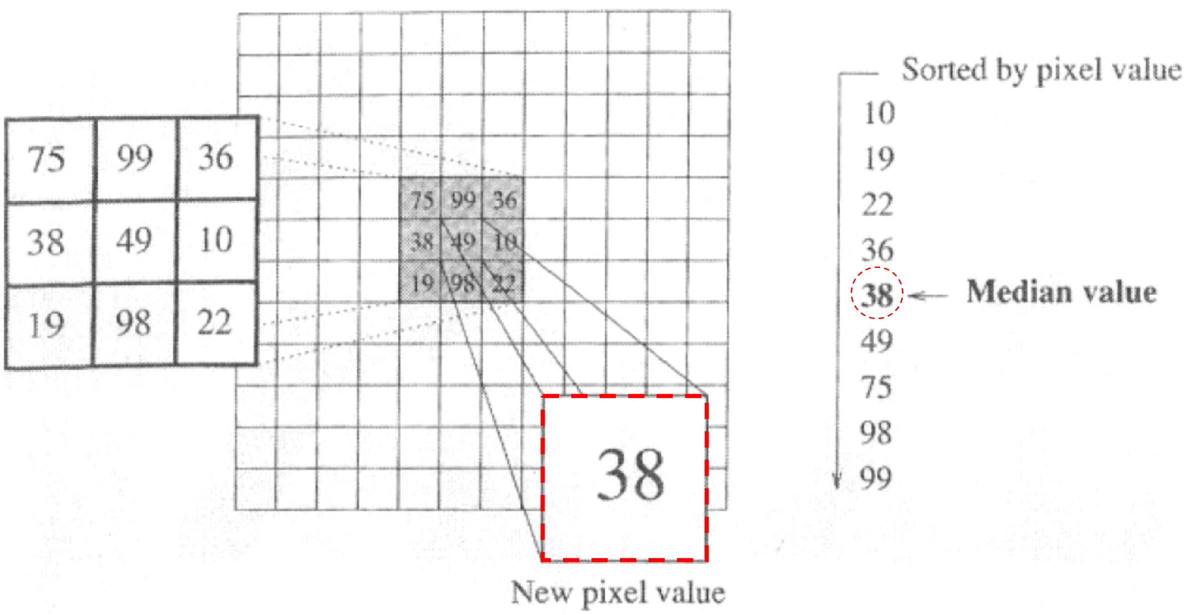
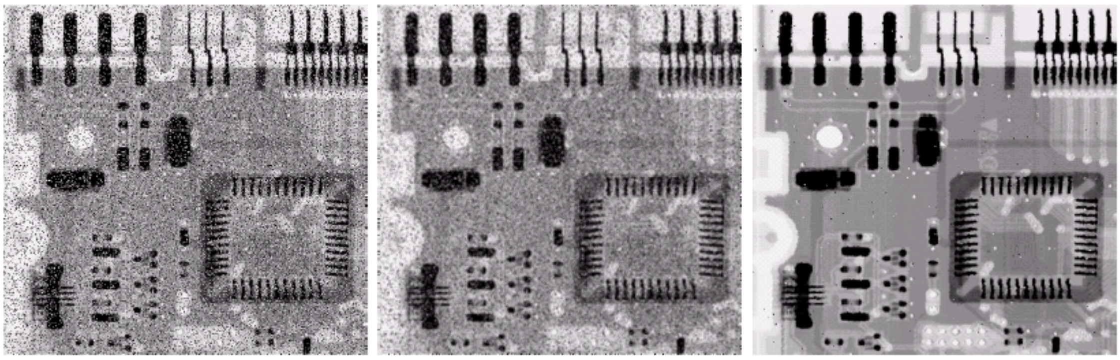


Image Enhancement

Non-linear Smoothing Filters – *median filter*

- For certain types of random noise, median filters provide excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filters of similar size.
  - Median filters are particularly effective in the presence of *impulse noise*, also called *salt-and-pepper noise* because of its appearance as white and black dots superimposed on an image.



X-ray image of circuit board corrupted by salt-and-pepper noise | Noise reduction with a 3x3 averaging filter | Noise reduction with a 3x3 median filter

Image Enhancement

Spatial Filtering – Image sharpening

- The principal objective of sharpening is to highlight fine detail in an image or enhance detail that has been blurred, either in error or as a natural effect of a particular method of image acquisition.

- ***The Unsharp Masking*** approach*
  - A process that has been used for many years by the printing and publishing industry to sharpen images consists of subtracting an unsharp (smoothed) version of an image from the original image.

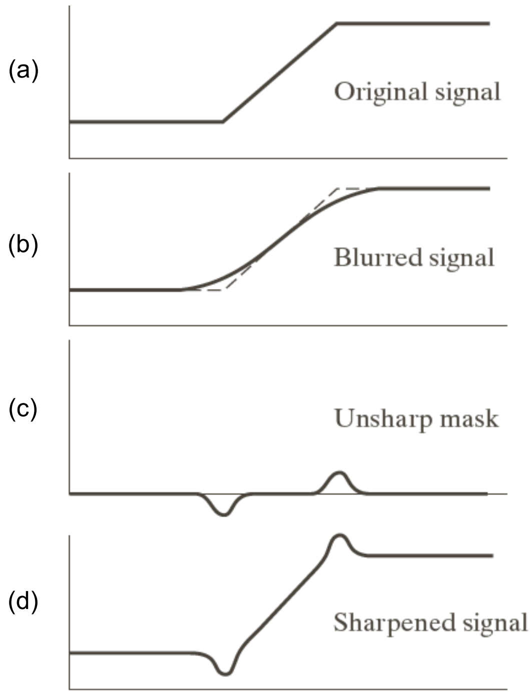
```
1.  Blur the original image
2.  Subtract the blurred image from the original
    (the resulting difference is called the mask.)
3.  Add the mask to the original
```

  - Variations of the unsharp mask can be created using different sizes of masks or types of blur / smoothed masks.

* The name "*Unsharp Masking*" is misleading because this filter actually sharpens images. The name is derived from the way the mask is constructed.

Image Enhancement

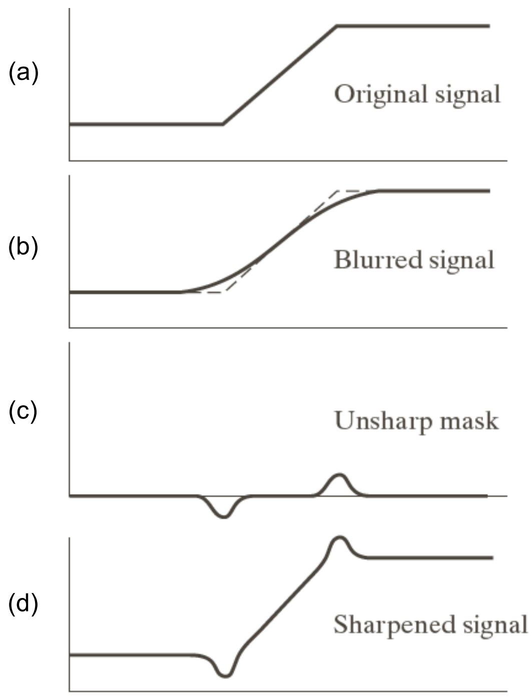
1-D illustration of the mechanics of the *Unsharp Masking* approach



- The intensity profile in Fig-(a) can be interpreted as a horizontal scan line through a vertical edge that transitions from a dark to a light region in an image.
- Fig-(b) shows the result of smoothing, superimposed on the original signal (shown dashed) for reference.
- Fig-(c) is the unsharp mask, obtained by subtracting the blurred signal from the original.
- Fig-(d) is the final sharpened result, obtained by adding the mask to the original signal. The points at which a change of slope in the intensity occurs in the signal are now emphasized (sharpened).

Image Enhancement

1-D illustration of the mechanics of the *Unsharp Masking* approach



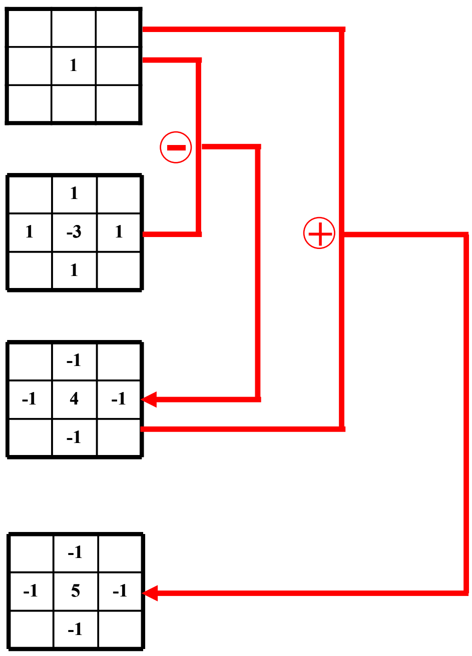


Image Enhancement

Spatial Filtering – Image sharpening

Original image

Result of a typical *Laplacian* filter mask

-1	-1	-1
-1	9	-1
-1	-1	-1

A typical image sharpening filter mask used to implement the linear *Laplacian* operator

Appendix

- Spatial filtering for image segmentation:

  Edge-detection

Appendix Spatial Filtering for Image Segmentation – *edge-detection*

The original image with some very high-contrast transition areas

.6	.6	.9	.9	.9
.6	.6	.9	.9	.9
.6	.6	.9	.9	.9
.6	.6	.6	.9	.9
.6	.6	.6	.6	.9

3 x 3 kernel

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

Edge-detection Convolve

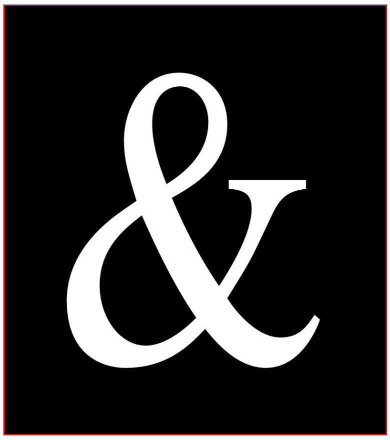
	0	.9	0	
	0	1	.3	
	0	0	.9	

The resulting image with detected edge

We apply a simple 3x3 kernel matrix as a mask and it is laid over the source pixel, where the source pixel is aligned with the center of the matrix and each of the eight neighboring pixels is aligned with its positional counterpart. We then multiply each pixel by the coefficient with which it is aligned. In this example, the value of every neighboring pixel is multiplied by -1, and the source pixel itself is multiplied by 8. Now we add together all of these values to get a new number. This number becomes the value of our new pixel in the destination image. (Number above 1.0 or below 0 are clipped.) This process is repeated for every pixel in the source image until we have created a new image, point by point.

The filter we applied here is one that is designed to detect edges in an image. It produced a bright pixel wherever there was a transition area in the original image, and produced dark pixels wherever the source image has a constant tone.

Appendix Spatial Filtering for Image Segmentation – *edge-detection*

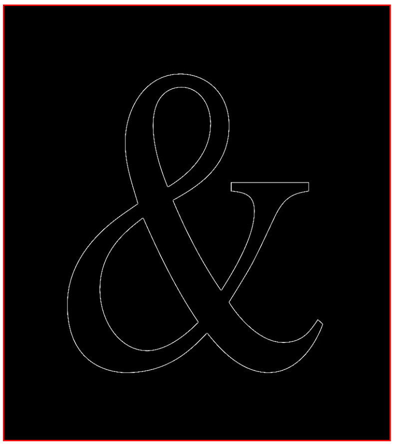


The original image

3 x 3 kernel

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

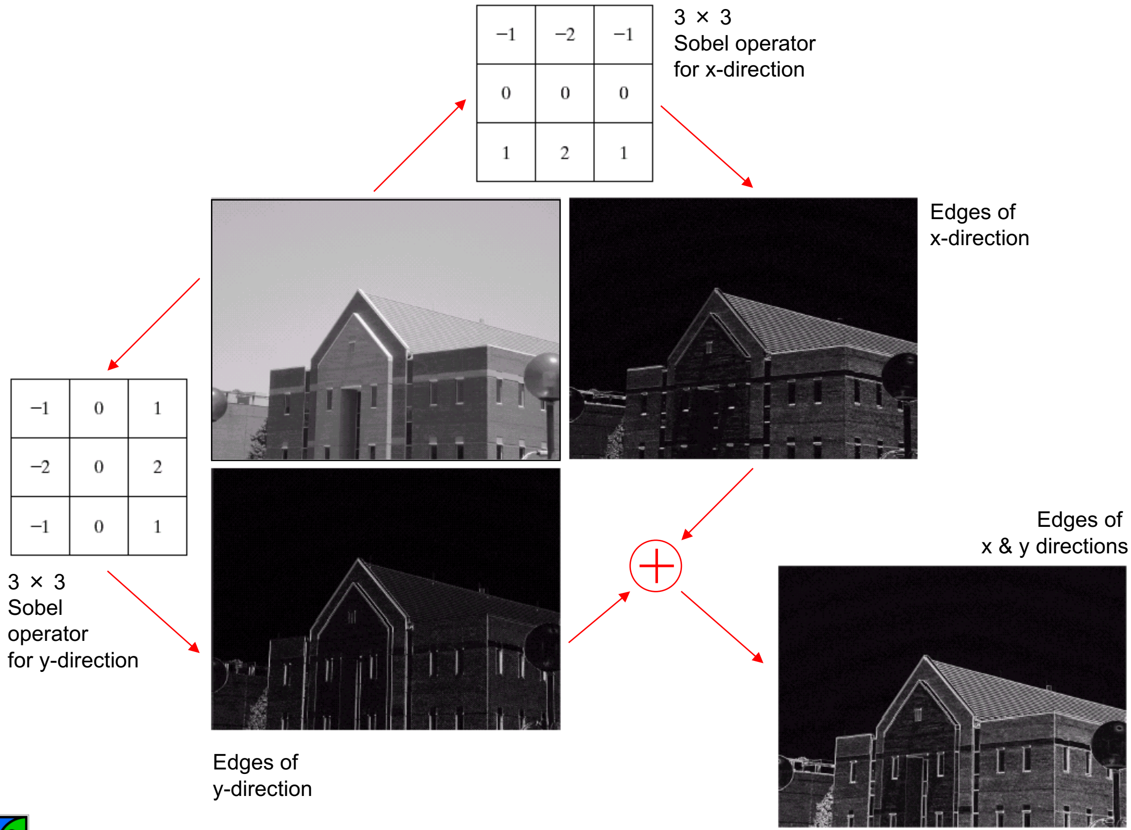
Edge-detection Convolve



The resulting image with detected edge

There are a large number of different convolution filters that are commonly used for processing images.

Appendix Spatial Filtering for Image Segmentation – *edge-detection*



Appendix Spatial Filtering for Image Segmentation – *edge-detection*