# Lecture 1: introduction.

Computer imaging: the acquisition & processing of visual information by computers.

* Info is received by computers & human visual system (Eyes).

It has two categories:

* Computer vision: the processed output images are used by computers.
  + Image analysis is one of its sub-fields.

It involves the examination of the image’s data to aid in solving visual problems.

It includes two topics:

* + - Feature extraction: acquiring higher level image information.
    - Pattern classification: taking the higher-level info & identifying objects within it.
* Image processing: the output images are used by humans.
  + Algorithms that alter images to create now ones.

## Digital image processing.

* Image processing.
  + Image in => image out.
* Image analysis.
  + Image in => measurements out.
* Image understanding.
  + Image in => high level description out.

Image: a two-dimensional function F(x, y).

* Intensity: The amplitude of F.
* x & y are spatial coordinates.

Digital image: a finite number of elements, each one has a location & a value.

* Picture element (pixel): the smallest element of an image.
  + Sometimes called: pels || image elements.
* Image formats:
  + 1 sample per point (B&W || Grayscale).
  + 3 samples per point (Red, Green, Blue).

Digital image processing focuses on:

* Improvement of image info for human understanding.
* Processing image data for storage, transmission, and representation.

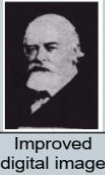
Digital image processing levels:

* Low-level (image processing):
  + Text

    Description automatically generatedInput: an image.
  + Output: an image.
* Mid-level (image analysis):
  + Input: an image.
  + Output: the image’s attributes.
* High-level (computer vision):
  + Input: image attributes.
  + Output: understanding of that image.

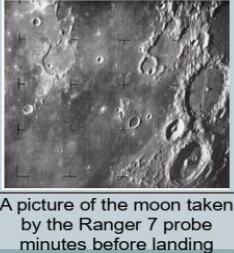
## A picture containing text Description automatically generatedHistory.

Early 1920’s:

* The newspaper industry used a digital image.
* Bartlane cable picture transmission service immerged.
* Images were transferred using submarine cables.

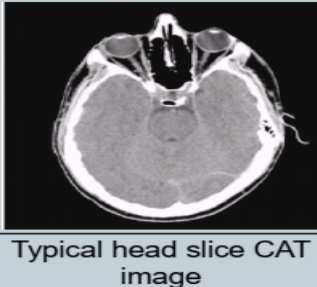
Mid to late 1920’s:

* Improvements to the Bartlane system (higher quality).
  + Increased the number of tones in image.

1960’s:

* Improvements in computing technology.
* Digital image processing was used to improve images of the moon.

1970’s:

* Digital image processing enters the medical field.
* Godfrey & his colleagues won a Nobel prize for their CAT scans.
  + Computerised Axial Tomography.
  + Combines many x-ray images.
  + Can generate 3d scans too.

1980’s – today:

* Digital image processing expanded.
  + Graphical user interface, text

    Description automatically generatedImage enhancement/restoration.
  + Artistic effects.
  + Medical visualization.
  + Industrial inspection.
  + Law enforcement.
  + Human computer interface.

## Image enhancement.

* One of the most used DIP techniques.
  + Improves quality & removes noise.
* Used with MRI scans.
* Used to improve some images sent by the Hubble telescope.

Diagram

Description automatically generatedFace recognition & gesture recognition is used to make human computer interfaces more natural.

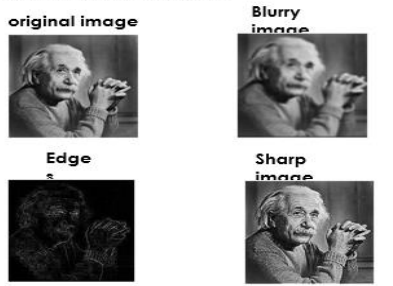
## Key stages in digital image processing.

1. Image Acquisition.
2. Image enhancement.
3. Image restoration.
4. Morphological processing.
5. Segmentation.
6. Object recognition.
7. Representation & description.
8. Color image processing.
9. Image compression.

# Lecture 2: Key Stages.

## Image sharpening & restoration:

* A picture containing text

  Description automatically generatedRefers to the process of changing Images for various purposes.
  + Zooming.
  + Blurring.
  + Sharpening.
  + Gray to color conversion.
  + Edge detecting.
  + Image retrieval & recognition.
* Image enhancement: improving an image.
* Image restoration: removing distortion from an image.

## Image:

A two-dimension array of square pixels, arranged in columns & rows.

* Table

  Description automatically generatedPixels are displayed horizontally row by row.
* Each pixel is assigned a value (black, white, gray, color).
* A picture containing text

  Description automatically generatedMatrix representation: Matrix {i, j}, value A {i, j}.
* Image representation: Pixel {i, j}, value A {i, j}.

### 2D & 3D:

* An image is a representation of our 3D world.
* An image is represented by a 2D array.
  + Because it doesn’t have any depth.

### Bit depth (bits per pixel || bpp):

* refers to the number of bits per pixel in an image.
* Or the number of bits representing each pixel.
* The greater the bpp, the greater the number of shades that can be represented.
  + Number of shades = 2bpp.
* The smallest bit depth (1 bit) is used on monochrome images.
* Gray scale uses 8 bits for each pixel.
* True color uses 24 bits for each pixel.

Image size depends on:

* Number of rows.
* Number of columns.
* Bit-depth.
* Size (bits) = rows \* cols \* bpp.

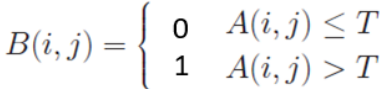
Image file formats:

1. Graphics interchange format (GIF).
   1. Connected to the WWW.
   2. Its standard is limited to 8-bits.
2. Joint Photographic Experts Group (JPEG).
   1. This standard was created by a group with the same name.
   2. Takes advantage of the human eyes limitations to reduce size.
   3. This limitation is more visible in color.
3. Portable Network Graphics (PNG).
   1. Supports up to 48 bits of color information (large).
4. Bitmap (BMP).
   1. Used in Microsoft OS (windows).
   2. Uncompressed image format.

## RGB to Grayscale conversion.

1. Average method.
   1. Grayscale = average of the three colors.
   2. Grayscale = (R + G + B)/3.
   3. Image is turned into a black image not gray.
   4. Most common grayscale conversion method.
   5. Fast & simple.
2. Lightness (de-saturation) method.
   1. Averages the most & least prominent colors.
   2. Grayscale = midpoint between maximum & minimum of colors.
   3. Grayscale = (max(R,G,B) + min(R,G,B))/2.
   4. Image is flatter & softer.
3. Weighted (luminosity) method.
   1. A more sophisticated version of the average method.
   2. Weighs each color based on how our eyes perceive it.
   3. Grayscale = weighted average.
   4. Grayscale = ((0.3 \* R) + (0.59 \* G) + (0.11 \* B)).
   5. Image is brighter than the average method image.

## Grayscale to Black & white conversion (Thresholding).

* The simplest method of image segmentation.
* Used to create binary images from a grayscale image.
* Replaces pixels:
  + With a black pixel if image intensity (i) is less/equal than constant (T).
  + With a with pixel if image intensity (i) is greater than constant (T).
* Default value for constant T = 128.

## Image zooming:

Zooming: enlarging a picture.

* Details become more visible & clearer.
* Pre-processing zoom: zooming before taking a picture.
  + Involves hardware & mechanical movement.
* Post processing zoom: zooming after taking a picture.
  + Uses algorithms that manipulate pixels to zoom.

### Optical zoom & digital zoom.

* Optical zoom (True zoom):
  + Achieved by moving the camera’s lenses.
  + Results are better than digital zoom.
  + Objects appear closer to the camera.
  + The lens gets physically extended to zoom/magnify.
* Digital zoom:
  + Image processing within the camera.
  + The center of the image is magnified, and the edges are cropped.
  + Objects appear closer to you.
  + Quality drops.

### Zooming methods.

1. Table

   Description automatically generatedPixel replication (nearest neighbor interpolation).
   1. Replicates the neighboring pixels (increases number of pixels).
   2. Table

      Description automatically generatedEach pixel is replicated N times row & column wise.
   3. Row zooming:
      1. Table, calendar

         Description automatically generatedCopy each pixel to a new adjacent cell.
   4. Column zooming:
      1. Copy each pixel to a new cell below it.
   5. New size =
      1. New row (original rows \* zooming factor).
      2. New column (original columns \* zooming factor).

Advantage & disadvantage:

* Very simple (adv).
* Very blurry result (dis).
* Image gets blurrier the higher the zooming factor.

Table

Description automatically generated

1. Zero order hold (zoom twice).
   1. Diagram

      Description automatically generatedCan only zoom twice.
   2. Places a new cell between two adjacent cells.
   3. New cell value = sum of two cells/2.
      1. Table

         Description automatically generatedDone row & column wise.
   4. New size =
      1. New row ( (2 \* original row) -1).
      2. New column ( 2 \* original column) -1).

Advantage & disadvantage:

* Doesn’t create blurry pictures (adv).
* Can only run on the power of 2 (dis).

1. A picture containing table

   Description automatically generatedZooming K times.
   1. K is the zooming factor.
   2. Steps:
      1. Text

         Description automatically generatedTake two adjacent cells.
      2. Sub the smaller from the greater.
      3. Divide the output by K (OP).
      4. Add the smaller value.
      5. Place cell.
      6. Add OP and place cell (repeat K-1 times).Table

         Description automatically generated
      7. Sort values after all new cells are inserted.
   3. New size =
      1. New row (K (original row -1) +1).
      2. New column (K (original column -1) +1).

Advantage & disadvantage:

* Can compute zoom of any factor (adv).
* Images are less blurry (adv).
* High cost of computation because of the sorting (dis).

# Lecture 3: image Enhancement.

## Basic info:

Total number of pixels = num of rows \* num of columns.

Pixel resolution: total number of pixels in a digital image.

Megapixels: equal to one million pixels.

* column pixels X row pixels / 1 million.

## Image enhancement properties:

* emphasizes the edges (boundaries).
* Improves visual appearance/content.

## Point operations.

* Simple & efficient class of image enhancement techniques.
* Value of each pixel is recalculated independently.
* Involves Adjustment of brightness & contrast.
* Applied to bad images (insufficient lighting || contrast).

## Image enhancement measures (parameters):

* brightness.
* Contrast: difference between maximum & minimum.
* Histogram: graph of the frequency of occurrence of data.

### Brightness & contrast.

* Overall brightness can be altered by adding/subbing (B) from each pixel.
  + G(x, y) = F(x, y) + B.
  + Increases if (B>0), decreases if (B<0).
* Overall Contrast can be altered by multiplying each pixel by (a).
  + G(x, y) = a \* F(x, y).
  + Increases if (a > 1), decreases if (a < 1).

### Negation/inversion (negative image).

* Subtracts each pixel from the maximum pixel value.
  + G(x, y) = 255 – F(x, y).
* Usefulness:
  + Displaying medical images.
  + Producing negative prints of images.

### Chart, histogram Description automatically generatedImage Histogram:

* The frequency of occurrence of gray levels in an image.
* Histogram of 8-bit image is a table of 256 entries.

### Intensity level slicing.

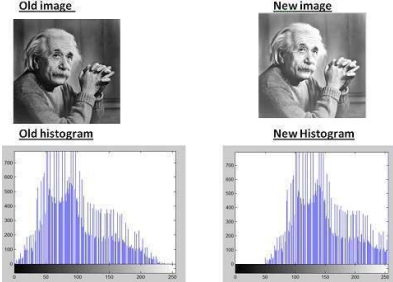
* Segments certain gray level regions from the rest of the image.
* A picture containing text, watch, clock, gauge

  Description automatically generatedUseful if region is within a gray level range.
* Implementations:
  + Keeping original pixel values of region and setting rest to zero.
  + A picture containing text, watch, clock

    Description automatically generatedReplacing pixel values by the highest value & keeping the rest in their original value.
* Managed as a thresholding operation with multiple boundaries.

### Histogram sliding.

Increasing

* Graphical user interface, chart, histogram

  Description automatically generatedShift histogram right || left.
* The image gets clearer.

Decreasing

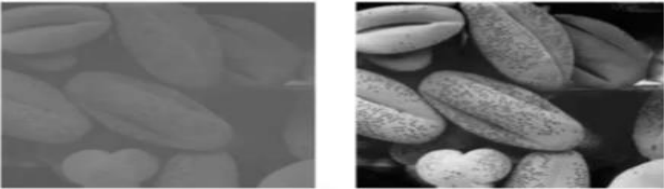
* can be used to represent increases & decreases in brightness.
* Histogram stretching: used to increase contrast.
* Histogram equalization: used to enhance contrast.
  + S = (L/n) \* T(r)-1.

## Image enhancement domains:

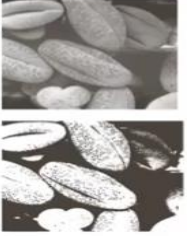
* Spatial domain.
* Transform domain.

# Lecture 4: contrast stretching & thresholding.Table Description automatically generated with medium confidence

## Contrast stretching (Normalization):

* One of the simplest piecewise linear functions.
* Used to increase the dynamic range of gray levels.
* Bright pixels get brighter & dark pixels get darker.
* Causes of low contrast:
  + Poor illumination/lighting.
  + Lack of dynamic range in image sensor.
  + Wrong settings in lens.
* Equation:
  + Graphical user interface

    Description automatically generated with medium confidenceI: input intensity of pixel.
  + INew: new intensity.
  + Min & Max: input intensity range.
  + newMin & newMax: new input intensity range.



## Diagram Description automatically generatedStretching & thresholding:

* thresholding is a special case of stretching.
  + Intensity doesn’t change.
* In contrast stretching pixels have a wide range.
* In thresholding pixels are 1 || 0 (binary image).

## Histogram equalization:

* Grayscale images have one histogram & Colored images have three.
* Concentration is on the low side in Dark image histograms.
* Chart

  Description automatically generatedConcentration is on the high side in bright image histograms.
* Concentration is in the middle in low contrast images.
* No concentration in high-contrast images (wide range).
* Colored images:
  + hR (x) = number of pixel in I(:,:,1) with intensity value x.
  + hG (x) = number of pixel in I(:,:,2) with intensity value x.
  + hB (x) = number of pixel in I(:,:,3) with intensity value x.

## Color to Gray.

* Lightness.
  + Lightness = ½ \* (max(R,G,B) + min(R,G,B)).
* Luminosity.
  + Luminosity = 0.21 \* R + 0.72 \* G + 0.07 \* B.
* Average.
  + Average brightness = ( R + G + B)/3.

## Probability mass function (PMF).

PMF: the probability of each number in a set (frequency of each number).

Methods to calculate PMF:

* From a matrix.
  + Take each value and count the amount of its appearance in the matrix.
  + Chart, bar chart, histogram

    Description automatically generatedThe probability = amount / total.
* From histogram.
  + Find the max length of each bar.
  + Probability = max length / total length.

## Cumulative distributive function (CDF).

Diagram

Description automatically generatedCDF: calculates the cumulative sum of all values that are calculated using PMF.

* The first value in the histogram doesn’t change.
* And the other values are the sum of the previous ones.

## PMF & CDF in histogram equalization.

* The first & second step of histogram equalization are PMF & CDF.
* PMF calculates the probability of each pixel.
* CDF calculates the cumulative sum of the values.
* The CDF is then multiplied by (num of levels – 1) to find the new pixel intensities.
* Equalization doesn’t always increase contrast.

# Lecture 5: Image Enhancement 2.

## Spatial filtering.

* Spatial filtering: technique used to modify the spatial resolution of an image.
* A picture containing text, screenshot, font, diagram

  Description automatically generatedA filter is a m \* n mask (weights).
* These weights are selected based on what the filter is intended to do.

### Correlation:

* The sum of products of two signals.
* Used to characterize the statistical dependencies between two signals.

### A diagram of a system Description automatically generated with low confidenceConvolution:

* A mathematical way to combine two signals to generate a third.
* Used to linearly filter a signal.
* A picture containing font, text, handwriting, line

  Description automatically generatedNew signal = sum of products of two signals.
  + The mask is flipped 180 degrees.

## Neighborhood operations.

* A picture containing square, number, screenshot, text

  Description automatically generatedThe value of any specific location (x, y) is the result of applying an operation on its neighboring pixels.
* Factors:
  + Operator (filter, mask, weight matrix).
  + A blue arrow pointing to the right

    Description automatically generated with low confidenceNeighborhood size.
* Operations:
  + A screenshot of a game

    Description automatically generated with medium confidenceMin: pixel value = the minimum value of its neighboring pixels.
  + Max: pixel value = the maximum value of its neighboring pixels.
  + Average: pixel value = the average value of its neighboring pixels (sum/size).
  + Median: pixel value = midpoint value in the Ordered set.
    - It’s sometimes better than avg.

### Dealing with image borders.

* A picture containing square, rectangle, line, screenshot

  Description automatically generatedAt these edges pixels don’t have a full neighborhood.
* Methods:
  + Exclude missing pixels.
    - Doesn’t work with all filters.
    - Can add extra code & slow down processing.
  + Pad the image.
    - Used with all white || black pixels.
  + Replicate border pixels.
  + Truncate the image.
  + Allow pixels to wrap around the image.
    - Can cause some strange image artifacts.

## Smoothing filters (SKIP THIS PART !!!!!!!).

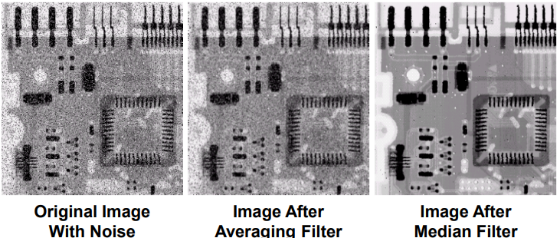
* Used for blurring & noise reduction.
* Used in removing small details (before object extraction).
* Used to fill small gaps in lines & curves.
* Used in linear & non-linear filters.

### A picture containing font, number, clock, symbol Description automatically generatedAveraging filter (low-pass filter).

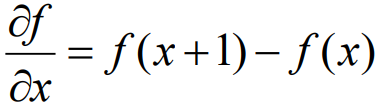
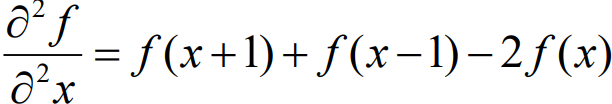
* A picture containing number, square, screenshot

  Description automatically generatedAmong the simplest linear filtering operations.
* Response = average of all pixels in neighborhood.
* Weighted average: different pixels in the same neighborhood can have different weights.
  + Pixels closer to the center are more important.
  + More effective for smoothing.
* Highlighting gross detail: mask size determines the relative size of the objects blended.
* Usage:
  + Resulting image has reduced sharp transitions.
  + Used in removing noise & reducing irrelevant details.
* Edges are blurred (disadvantage).

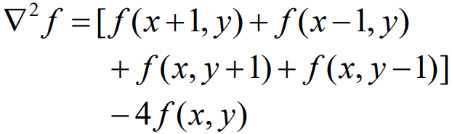
### Non-linear filters.

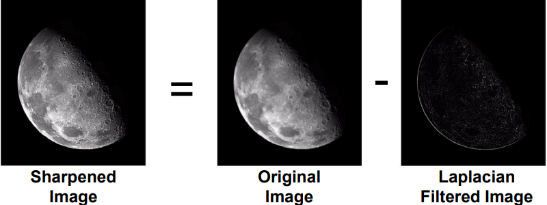
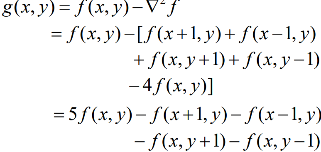
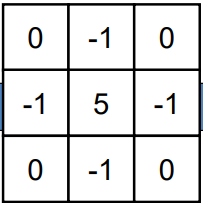
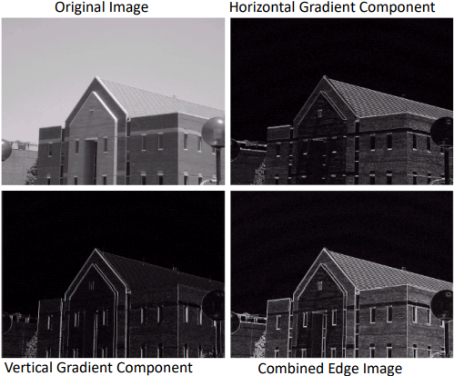
* Filters where the response is based on the ordering of the pixels.
* Median filter: sorts pixels, determines the median & assigns it to the center pixel.
  + Changes pixels with distinct intensities to be like their neighbors.
  + Eliminates isolated clusters of pixels that are light || dark (relative to their neighbors).

## Sharpening filters (SKIP THIS PART!!!!!).

* Based on spatial differentiation.
* Measures the rate of change of a function.
* Uses first & second derivatives.
* Objectives:
  + Highlights edges: by highlights intensity transitions.
  + Enhance details: by reducing blurriness.
* First derivative:
  + The difference between subsequent values.
  + Measures the rate of change.
* Second derivative:
  + Accounts for the values before & after the current one.
  + Value’s sign changes at the start & at the end.
  + More useful than the first derivative.
  + Has a stronger response to fine detail.
  + A picture containing square, screenshot, line, rectangle

    Description automatically generatedSimpler to implement.



* Second derivative (Laplacian):
  + Used in edge detectors.
  + Can be represented by a matrix.
  + Subtracting the result from the original image results in a sharpened image.
    - Depends on the sign of the mask.
* Second derivative (subtracted Laplacian):
  + Subtracting the Laplacian image from original in one step.
  + The filter can be used for sharper images.
* First derivative (gradient):
  + Used for edge detection.
  + X derivative = third row – first row.
  + A picture containing square, number, rectangle, line

    Description automatically generatedY derivative = third column – first column.
  + Its matrices (Sobel operators).
* First & Second comparison.
  + First derivative:
    - Produces thicker edges.
    - Has a stronger response to gray level step.
  + Second derivative:
    - Produces a double response at step changes in gray level.
    - Has a stronger response to fine details.

# Lecture 6: Steganography.

Covert communication: two parties communicating with each other without anyone else knowing.

Its branches:

* Cryptography:
  + The process of scrambling the message to hide it.
  + Plaintext: the original message.
  + Cipher text: the encrypted message.
  + Encryption: the process of taking plaintext & converting it into cipher text.
  + Decryption: the process of taking cipher text & converting it into plain text.
* Information hiding.
  + Types:
    - Watermarking: placing imprints in a document to prove its authenticity.
    - Steganography.

## Steganography.

* The art & science of hiding info in a picture.
* It mustn’t cause suspicions about the medium (it fails if that happens).
* A picture containing text, screenshot, line, font

  Description automatically generatedDerived from ‘steganos’ (covered) & ‘graphia’ (writing).
* Cover image (carrier): the original image where the message will be embedded.
* Payload (message): the secret message that will be embedded.
* Stego image: the image after the message has been embedded.
* Hiding capacity: the size of info that can be hidden in the cover image (without reducing the quality).

### Its goals:

* Imperceptibility:
  + the visual difference between the two images.
* Capacity:
  + the max amount of data that can be embedded.
* Security:
  + the resistance to the attackers.

### Features:

* Low availability of encryption services.
  + Motivated people to study its methods.
* Images don’t look suspicious.
* Some implementations use a secret key (problem for attackers).

### Steganography categories:

* Substitution system techniques.
* Transform domain techniques.
* Spread spectrum techniques.
* Statistical method techniques.
* Distortion techniques.
* Cover generation techniques.

Substitution system techniques.

* Replaces unneeded bits of the cover with bits from the message.
* Several tools use the Least-significant bit (LSB) method.
  + Cover can be (image, audio, video).
  + Takes advantages of the wasted bits in covers.
  + Most significant bit: the bit that causes the most change in the cover.
    - the left most bit (in gray images).
    - The red value (in color images).
  + Least significant bit: the bit that causes the least change in the cover.
    - the right most bit.
    - The blue value.
* LSB steps:
  + Locate the next pixel.
  + Find its least significant bit (right most bit).
  + Locate the next message bit (start left to right).
  + Replace the pixel bit with the message bit (some implementations use XOR instead).
    - Ex. pixel bit = 1, message bit = 0 => new pixel bit = 0.
    - Ex. pixel bit = 1, message bit = 1 => new pixel bit = 1.
  + Repeat until there are no message bits left.

# Lecture 7: