# IVP Midsem Notes

## Imaging System

- H/W + S/W to capture, process, and display images
- array of sensors to convert light signals to electrical signals
- major subsystems
    - optical sys - lens + paerture + shutter
    - image sensor - converts optical img to electrical signal
    - DSP sys - processes captured signal for enhancement, compression, and storage
        - functions
            - image enhancement - noise reduction, contrast adjustment, sharpening, etc.
            - compression - JPEG, PNG, etc.
                - lossy - by removal of some data; JPEG
                - lossless - by removal of unnecessary metasat wihtout any discernible loss in quality; PNG
- sampling theorem
- sampling
    - process of convering a continuous time signal to a discrete tiem signal by measuring its amplitude at regular intervals
    - Fourier Analysis is performed to find the component sin waves of a signal
    - sampling theorem - for a band-limited signal of max freq x Hz, atleast 2x samples must be taken to rep the signal

> band-limited samples are those that can be completely reconstructed from its samples

- quantization
    - process of mapping continous amplitude/intensity vals into discrete levels
    - crucial step in img compression
    - number of levels available for quantization depends of bith depth n = $2^n$
    - more levels reduce quanitsation error
    - type
        - uniform - equal-sized intervals
        - non-uniform - unequal-sized intervals
    - types of resolution
        - spetial resolution - dependent on number of sensors
        - intensity resolution - dependent of accurate representation of intensity variations

$$Q(i) = \text{round}\left(\frac{i}{L} \times N\right)$$

## Gaussian Distribution

- PDF

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$f(x)$ = probability density function
$\sigma$ = standard deviation
$\mu$ = mean

- higher std. dev. makes a wider surve and lower std. dev. makes a narrower curve
- CDF

$$F(x) = P(X \le x) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right]$$

$$\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$$

- normalized distribution; mean = 0, std. dev. = 1

$$\sigma = \sqrt{\frac{\sum (x_i - \mu)^2}{N}}$$

$\sigma$ = population standard deviation
$N$ = the size of the population
$x_i$ = each value from the population
$\mu$ = the population mean

> standard deviation represents the amount of variation of a variable about its mean

## Image Filtering

- types
  - point processing
    - operates on each pixel independently
    - brightness, contrast, etc.
    - Gamma correction
      - correction applied to a pixel based on the characteristics of the display
      - non-linear operation used to adjust brightness and contrast of an image
      - mimics human vision in the sense that eyes are more sensitive to darker areas than bright ones

$$I_{\text{out}} = I_{\text{in}}^{\gamma}$$

where:

- $I_{\text{in}}$ = Input pixel intensity (normalized between 0 and 1)
- $I_{\text{out}}$ = Output pixel intensity
- $\gamma$ (gamma value) = Controls the brightness
  - $\gamma < 1$ **(e.g., 0.5): Brightens the image**
  - $\gamma > 1$ **(e.g., 2.2): Darkens the image**
  - $\gamma = 1$**: No change**

  - neighbourhood processing
    - modifies pixel based on surrounding pixels
    - smoothing, sharpening, blurring, etc.
    - linear(weighted sum) shift invariant(uniform application of filter across image) image filtering
      - each pixel is replaced by a linear combination of its neighboruing pixels
- filtering processes
  - convolution
    - flips kernel before applying it on the image
    - for a 2D kernel, it is transposed and applied on the image

$$g(x, y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} f(x+i, y+j) h(-i, -j)$$

where:

- $f(x, y)$ is the input image.
- $h(i, j)$ is the filter kernel.
- $g(x, y)$ is the output image.

🔹 **Used in:**

- Edge detection (Laplacian, Sobel).
- Blurring (Gaussian filter).
- Feature extraction (Gabor filters).

  - correlation

- doesn't flip kernel before application

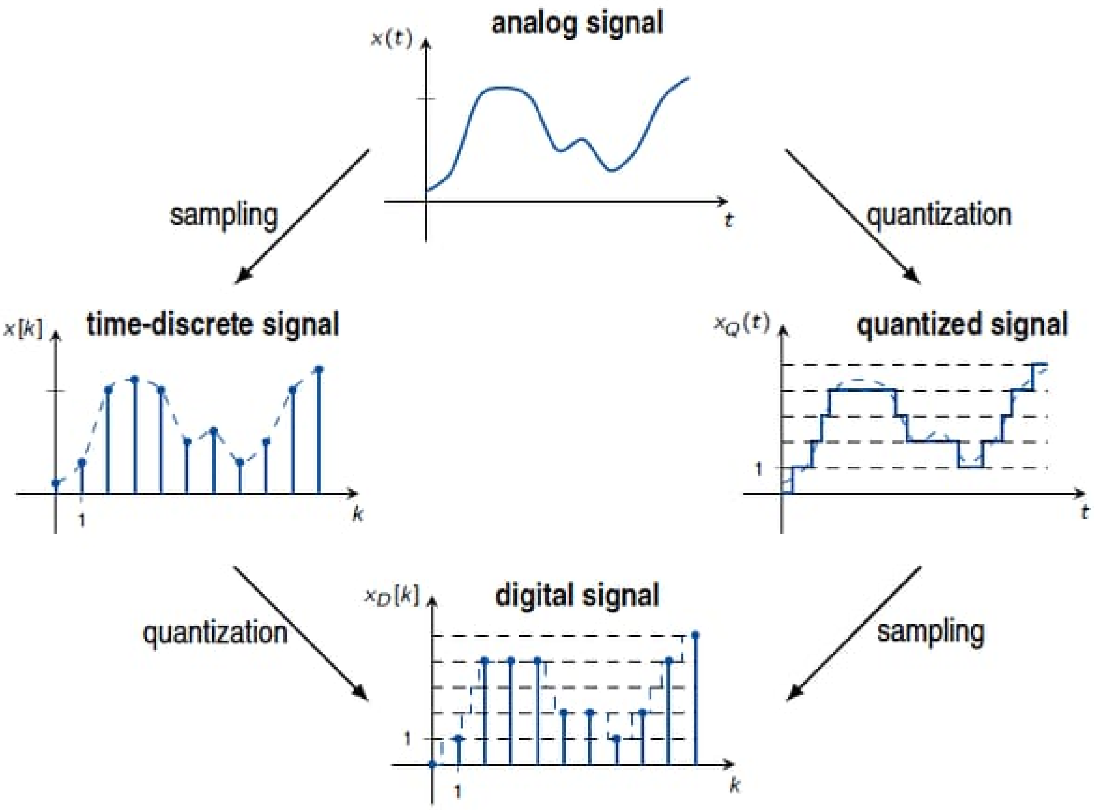
$$g(x,y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} f(x+i, y+j)h(i,j)$$

🔹 **Used in:**

- Template matching (Pattern detection).
- Feature detection (Corner detection).
- Similarity measurements.

💡 **Difference**:

- Convolution flips the filter kernel before applying it.
- Correlation directly applies the filter kernel.




- box/mean filter
  - for uniform filtering
  - properties
    - blurs image by averaging intensities
    - reduces noise
  - for smoothing and reducing noise before edge detection

$$h = \frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- Gaussian filter

    - theoretically infinite but we bound it to the max size of the kernel
    - we usually boud it as 2 - 3 * sigma
- for smooth filtering
    - properties
        - smooths noise while preserving edges
        - more natural blurring effect

$$h(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

    - used for denoising images

$$h = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

- misc
    - zero padding is performed at the edges of images to apply filters; no. of rows/columns to be zero-padded on either side of the input image is given by no. of rows/columns in the kernel - 1

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 25 | 100 | 75 | 49 | 130 | 0 | 0 |
| 0 | 0 | 50 | 80 | 0 | 70 | 100 | 0 | 0 |
| 0 | 0 | 5 | 10 | 20 | 30 | 0 | 0 | 0 |
| 0 | 0 | 60 | 50 | 12 | 24 | 32 | 0 | 0 |
| 0 | 0 | 37 | 53 | 55 | 21 | 90 | 0 | 0 |
| 0 | 0 | 140 | 17 | 0 | 23 | 222 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

    - separable filters
        - filters that can be represented as the product of a row and a column
        - for an M * M img and N * N kernel
            - non-separable - $M^2N^2$ convolutions
            - separable - $2M^2N$ convolutions
        - separable filters reduce the number of computations
    - noise
        - random variations in pixel intensity
        - obscures img details
    - rank of a matrix - no. of independent rows/columns

https://16385.courses.cs.cmu.edu/fall2020/lecture/filtering/slide_001