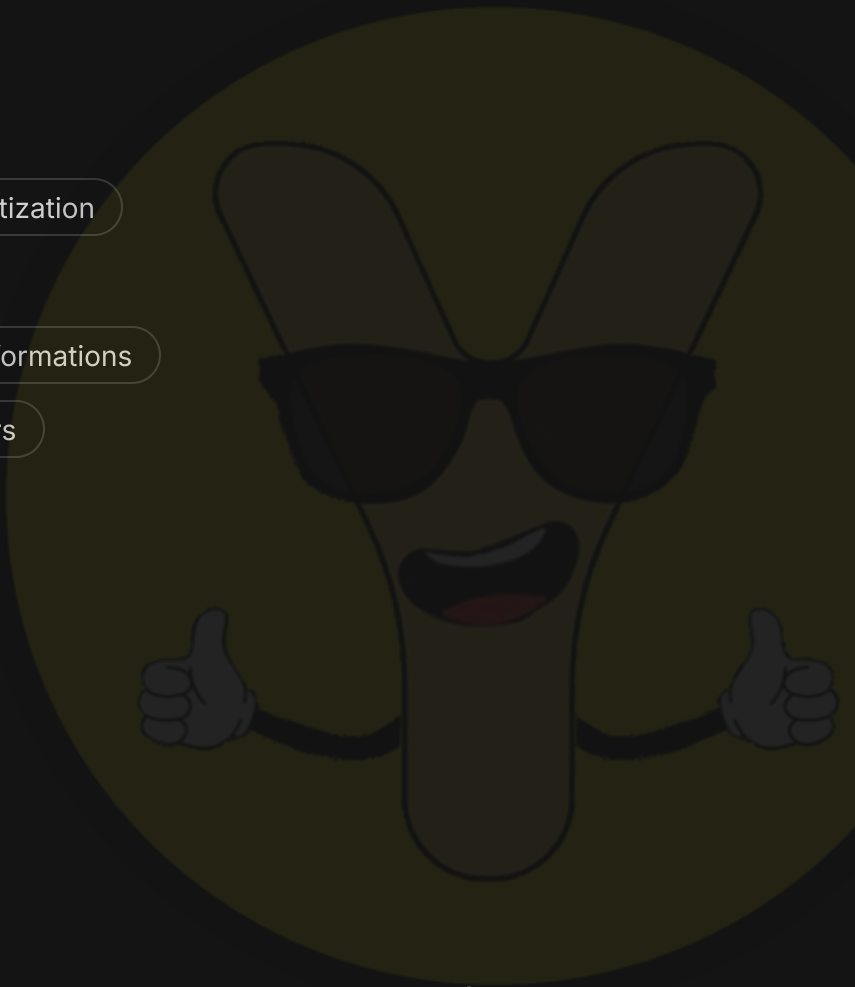
AKTU · 3RD YEAR · SEMESTER 6 · BCDS061

Image Analytics

Unit 1 — Fundamentals and Intensity Transformations

- Introduction
- Fundamental steps in Image Processing Systems
- Image Acquisition
- Sampling and Quantization
- Pixel Relationships
- Mathematical Tools Used in Digital Image Processing
- Some Basic Intensity Transformation Functions: Image Negatives
- Log Transformations
- Power-Law Transformations
- Histogram Processing
- Color Fundamentals
- Fundamentals of Spatial Filtering
- Smoothing Spatial Filters



DECODED BY

Chirag Jain

Founder, Ycotes · Gold Medalist in Artificial Intelligence, IIT Madras · taught 10,000+ students

theycotes.com

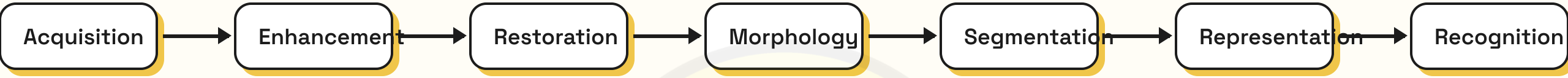
Introduction to Digital Image Processing

DIGITAL IMAGE PROCESSING (DIP)

Digital Image Processing involves manipulating digital images using computer algorithms to enhance visual information or extract useful features for automated analysis.

- A digital image is represented as a 2D matrix $f(x, y)$, where x and y are spatial coordinates.
- The intensity or gray-level value at any coordinate pair (x, y) is a discrete scalar quantity.
- Low-level processing inputs and outputs images (e.g., noise reduction, contrast enhancement).
- Mid-level processing inputs images and outputs attributes/features (e.g., segmentation, object recognition).
- High-level processing involves making sense of recognized objects (e.g., scene interpretation).

Fundamental Steps in Image Processing Systems



Sequential pipeline of fundamental steps in a standard Digital Image Processing system.

Overview of Image Processing Pipeline

Each fundamental step addresses a specific task in processing and understanding digital images.

- Image Acquisition
  - Capturing physical scenes and converting them into digital format using sensors.
- Enhancement & Restoration
  - Enhancement highlights features for human perception; restoration removes known degraded noise models.
- Color & Morphological Processing
  - Color processing handles RGB/HSV spaces; morphology extracts shape components using structural elements.
- Segmentation & Recognition
  - Segmentation partitions an image into objects; recognition assigns labels to extracted descriptors.

Image Acquisition

The process of sensing continuous physical energy and converting it into a discrete digital representation.

- Sensor Arrangement
  - Single sensing element (pinhole/line scan), line sensor arrays (flatbed scanners), and 2D arrays (CCD/CMOS cameras).
- Image Formation Model
  - An image is formed by illumination $i(x, y)$ reflecting off scene reflectance $r(x, y)$.
- Mathematical Expression
  - $f(x, y) = i(x, y) \cdot r(x, y)$, where $0 < i(x, y) < \infty$ and $0 < r(x, y) < 1$.
- Digitization Pipeline
  - Physical Energy continuous signal → Illumination & Reflectance → Sensor output voltage → Digitizer (ADC).

Sampling and Quantization

DIGITIZATION PROCESS

Digitization requires sampling continuous spatial coordinates into discrete grids and quantizing continuous intensity levels into discrete digital values.

- Spatial Sampling: Discretizing the spatial domain coordinates (x, y).
- Intensity Quantization: Discretizing the continuous amplitude range $f(x, y)$ into L levels.
- Image Resolution: Spatial resolution depends on sampling density; intensity resolution depends on bit depth k, where $L = 2^k$.
- Storage Requirement: Total storage bytes required for an $N \times M$ image with k bits per pixel is $b = N \times M \times k/8$.

Worked Example: Storage Calculation

Calculate the storage size in kilobytes (KB) for a gray-level image of size 512×512 pixels with 256 gray levels.

1. Identify spatial dimensions: $N = 512$, $M = 512$.
2. Determine bit depth k: Gray levels $L = 256 = 2^8 \implies k = 8$ bits per pixel.
3. Total bits required: $B = N \times M \times k = 512 \times 512 \times 8 = 2,097,152$ bits.
4. Convert to bytes: $\text{Bytes} = 2,097,152/8 = 262,144$ bytes.
5. Convert to Kilobytes (KB): $\text{KB} = 262,144/1024 = 256$ KB.

The total storage size required for the image is 256 KB.

Pixel Relationships: Neighbors and Connectivity

Spatial connections between pixels form the basis for region segmentation and boundary extraction.

- 4-Neighbors $N_4(p)$
  - The four orthogonal pixels surrounding pixel $p(x, y)$: $(x+1, y), (x-1, y), (x, y+1), (x, y-1)$.
- Diagonal Neighbors $N_D(p)$
  - The four diagonal pixels: $(x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1, y-1)$.
- 8-Neighbors $N_8(p)$
  - The complete set of surrounding pixels: $N_8(p) = N_4(p) \cup N_D(p)$.
- Connectivity Types
  - 4-connectivity, 8-connectivity, and m-connectivity (mixed connectivity to eliminate multiple path ambiguities).

Distance Measures Between Pixels

Distance Metric	Formula	Properties / Geometric Shape
Euclidean Distance (D_e)	$D_e(p,q) = \sqrt{(x-u)^2 + (y-v)^2}$	Forms circles of radius r around pixel p.
City-Block Distance (D_4)	$D_4(p,q) = \lvert x-u \rvert + \lvert y-v \rvert$	Forms diamonds; step count along 4-connected paths.
Chessboard Distance (D_8)	$D_8(p,q) = \max(\lvert x-u \rvert, \lvert y-v \rvert)$	Forms squares; step count along 8-connected paths.

Mathematical Tools Used in Digital Image Processing

Digital Image Processing relies on fundamental linear algebra, statistical, and set-theoretic operations.

- Array vs. Matrix Operations
  - Array operations are executed pixel-by-pixel; matrix operations follow strict matrix multiplication laws.
- Arithmetic Operations
  - Addition (image averaging for noise reduction), Subtraction (change detection), Multiplication/Division (shading correction).
- Set and Logical Operations
  - Union, Intersection, Complement; AND/OR masks applied bitwise on binary images.
- Spatial & Domain Transforms
  - Fourier Transform, Discrete Cosine Transform (DCT), Wavelet Transform for frequency domain analysis.

Basic Intensity Transformations: Image Negatives

IMAGE NEGATIVE TRANSFORMATION

An intensity transformation function that reverses pixel brightness levels, producing an output equivalent to a photographic negative.

- Mathematical Expression: $s = T(r) = (L - 1) - r$, where r is input intensity and s is output intensity.
- Intensity Level Range: Input intensity spans $[0, L - 1]$ (e.g., 0 to 255 for an 8-bit image).
- Primary Application: Enhancing white or light details embedded within dominant dark regions.
- Visual Effect: Pure black (0) maps to pure white ($L - 1$) and vice versa.

Log Transformations

Logarithmic mapping expands dark pixel values while compressing dynamic range of bright values.

- Formula
  - $s = c \cdot \log(1 + r)$, where c is a scaling constant and $r \geq 0$.
- Functionality
  - Maps a narrow range of low input intensity values into a wider range of output values.
- Dynamic Range Compression
  - Useful when image dynamic range spans several orders of magnitude (e.g., Fourier spectrum display).
- Inverse Log Transformation
  - Performs opposite operation: compresses low intensities and expands high intensities.

Power-Law (Gamma) Transformations

Power-law transformations adjust image contrast based on exponential parameter gamma.

- Mathematical Expression
  - $s = c \cdot r^{\gamma}$, where c and γ are positive constants.
- Gamma Correction ($\gamma < 1$)
  - Expands dark values, brightens image, enhances shadow detail (similar to log transform).
- Gamma Compression ($\gamma > 1$)
  - Compresses dark values, darkens image, enhances bright region highlights.
- Applications
  - CRT/LCD display monitor response correction, contrast adjustment in medical imagery.

Histogram Processing

The intensity histogram represents the frequency distribution of gray levels in an image.

- Definition of Histogram
  - Discrete function $h(r_k) = n_k$, where r_k is the k-th intensity level and n_k is pixel count.
- Normalized Histogram
  - $p(r_k) = n_k / MN$, giving an estimate of the probability density function (PDF).
- Histogram Equalization Goal
  - Transform intensity levels to yield a uniform histogram, maximizing global image contrast.
- Transformation Formula
  - $s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p(r_j) = \frac{L-1}{MN}\sum_{j=0}^{k} n_j.$

Worked Example: Histogram Equalization

Equalize a 3-bit image ($L = 8$) of size $64 \times 64 = 4096$ pixels with intensity counts n_k: [1000, 1500, 800, 500, 200, 96, 0, 0].

1. Compute normalized PDF $p(r_k) = n_k / 4096$: [0.244, 0.366, 0.195, 0.122, 0.049, 0.024, 0, 0].
2. Compute Cumulative Distribution Function (CDF): [0.244, 0.610, 0.805, 0.927, 0.976, 1.000, 1.000, 1.000].
3. Multiply CDF by $(L - 1) = 7$: [1.708, 4.270, 5.635, 6.489, 6.832, 7.000, 7.000, 7.000].
4. Round to nearest integer: $s_k = [2, 4, 6, 6, 7, 7, 7, 7]$.

Mapped output levels: $r_0 \to 2, r_1 \to 4, r_2 \to 6, r_3 \to 6, r_4 \to 7, r_5 \to 7, r_6 \to 7, r_7 \to 7.$

Color Fundamentals and Color Models

Color perception simplifies human vision and enhances feature separation in image analysis.

- Color Fundamentals
  - Human eye perceives color using cones (red, green, blue). Parameters: Brightness, Hue, Saturation.
- RGB Color Model
  - Additive model used in hardware displays; primary colors Red, Green, Blue form a unit cube.
- HSI / HSV Color Model
  - Decouples intensity/value from color info (Hue & Saturation); matches human visual perception.
- CMY / CMYK Model
  - Subtractive model used in color printing (Cyan, Magenta, Yellow, Key/Black).

Fundamentals of Spatial Filtering

Spatial filtering directly alters image pixel values using neighborhood mask operations.

- Spatial Filter Concept
  - Consists of a spatial neighborhood subimage (mask, kernel, filter) and an operation.
- Mechanics of Filtering
  - Moving a mask of size $m \times n$ (typically odd: 3×3, 5×5) from pixel to pixel.
- Linear Spatial Filtering
  - Output pixel is a weighted sum of input pixels: $g(x, y) = \sum\sum w(s, t) f(x + s, y + t)$.
- Correlation vs. Convolution
  - Correlation slides mask over image; Convolution flips mask 180 degrees before sliding.

Smoothing Spatial Filters

Smoothing filters reduce noise and blur sharp intensity transitions in an image.

- Linear Smoothing (Mean Filter)
  - Replaces pixel value with average of neighborhood; attenuates high frequency noise but blurs edges.
- Gaussian Smoothing Filter
  - Uses 2D Gaussian weighted distribution $w(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$; yields isotropic smooth blur.
- Order-Statistic Filters
  - Non-linear spatial filters based on ranking pixel intensities within the neighborhood grid.
- Median Filter
  - Replaces pixel with median intensity; highly effective at removing salt-and-pepper noise while preserving edges.

Sharpening Spatial Filters

Sharpening highlights intensity transitions, edges, and fine details using spatial derivatives.

- First Derivative (Gradient)
  - Highlights gross edges: $\nabla f = [\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}]^T$; magnitude $|\nabla f| = \sqrt{g_x^2 + g_y^2}$.
- Sobel & Prewitt Operators
  - Use 3×3 masks to calculate directional partial derivatives g_x and g_y.
- Second Derivative (Laplacian)
  - Isotropic 2D second derivative: $\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$.
- Image Sharpening Equation
  - $g(x, y) = f(x, y) + c[\nabla^2 f(x, y)]$, where $c = -1$ if center mask coefficient is negative.

Smoothing vs Sharpening Spatial Filters

Feature	Smoothing Filters	Sharpening Filters
Primary Goal	Noise reduction and image blurring	Edge enhancement and detail highlight
Frequency Domain Effect	Low-pass filtering (attenuates high freq)	High-pass filtering (attenuates low freq)
Mathematical Basis	Integration / Averaging	Spatial Differentiation / Derivatives
Typical Masks	All positive mask weights (e.g., 1/9 matrix)	Sum of mask weights equals 0 (zero DC response)
Edge Preservation	Blurs edges (except Median filter)	Accentuates edges and noise

Exam Corner: Unit 1 AKTU Questions

- **Q1** Explain the fundamental steps in a Digital Image Processing System with a neat block diagram.
- **Q2** Differentiate between Image Sampling and Quantization. Calculate the storage capacity needed for a 1024×1024 256-gray-level image.
- **Q3** Explain 4-neighbors, 8-neighbors, and diagonal neighbors of a pixel. Define m-connectivity with a suitable example.
- **Q4** Derive the mathematical expression for Histogram Equalization. Perform histogram equalization for a given 3-bit image matrix.
- **Q5** Explain Power-Law (Gamma) Transformation and Log Transformation with their applications and plots.
- **Q6** Differentiate between Correlation and Convolution in spatial filtering.
- **Q7** Explain the working of Median filter and compare its performance with Mean filter for salt-and-pepper noise.
- **Q8** Derive the Laplacian operator for image sharpening and explain how the Sobel mask detects directional edges.

Unit 1 done. You are exam-ready.

Revise the Exam corner once more, then open Unit 2 on theycotes.com. Every AKTU subject, unit by unit, free. Notes by Chirag Jain, Ycotes.

