Image Processing
Lecture 01: Introduction

Based on lecture by Shmuel Peleg

2025-26

Contents

1 Introduction to Image Processing & Computer Vision

- **Computational Photography:** Input is a scene, output is "Better Pictures".
- **Computer Vision (CV):** Input is pictures/video, output is "Info on Scene" (e.g., face recognition, 3D).
- CV is a subfield of **Artificial Intelligence (AI)** that focuses on learning from data.
- **History:** Field started ∼1964. Started to work ∼2014 with Neural Networks.
- **Applications:**
  - Face Recognition
  - Autonomous Driving (e.g., MobilEye)
  - Medical Diagnosis
  - Image Enhancement (e.g., Dehazing)
  - Video Synopsis (e.g., Briefcam)
- **Note:** Generative AI is **NOT** covered in this course.

2 Physics & Biology of Vision

2.1 Image Formation

- Light is emitted from a source, reflects from an object, and is sensed by an eye or camera.
- **Model:** $I = L \cdot r$
  - I = Intensity (sensed)
  - L = Light (from source)
  - r = Reflectance (of object surface)
- **Reflectance Types:**
  - **Specular (Shining):** Mirror-like, reflects at an equal angle.
  - **Diffuse (Matt):** Scatters light equally in all directions.

2.2 The Human Eye

- Light is focused by the Cornea and Lens onto the Retina.
- **Receptors on the Retina:**
  - **Rods ($\approx 10^8$):** Sense grayscale (B/W). Highly sensitive, work in low light.
  - **Cones ($\approx 10^7$):** Sense color (RGB). Less sensitive, work in bright light.

- We see color via three types of cones: Blue (high freq.), Green, and Red (low freq.).

3 The 3 Stages of Image Digitization

This is the fundamental process of converting the analog 3D world that we see into a digital file (a matrix of numbers) that a computer can store and manipulate.

3.1 Stage 1: Perspective Projection (Continuous 3D to 2D)

This is the physics and optics stage. It's how a 3D scene gets "flattened" onto a 2D surface, like the film in a camera or the retina in your eye.

- **The Model:** This process is modeled by the **Pinhole Camera** (also called *Camera Obscura* or "dark room").
- **The Physics:** In a pinhole camera, a barrier with a tiny hole (the "pinhole" or "Center of Projection") is placed between the 3D scene and a 2D image plane. By allowing only a single ray of light from each point in the scene to pass through this hole, a sharp, inverted image forms on the 2D plane.
- **The Math:** This transformation is described by the **perspective projection equations**. For a 3D point in the world (X, Y, Z), its 2D coordinate (x, y) on the image plane is calculated using similar triangles:

$$x = \frac{f}{Z}X \quad \text{and} \quad y = \frac{f}{Z}Y \tag{1}$$

  Here, f is the **focal length**, which is the distance from the pinhole to the image plane. This equation mathematically explains why objects farther away (a larger Z value) appear smaller in the image.
- **The Result:** The output of this stage is a **continuous, analog** 2D image. It's a real, physical projection of light, not yet a set of numbers.

3.2 Stage 2: Spatial Sampling (Continuous 2D to Discrete Grid)

This stage takes the continuous 2D image from Stage 1 and discretizes its **space**.

- **The Concept:** The continuous image plane is divided into a grid with a **finite number of pixels**.
- **The Mechanism:** In a digital camera, this is done by a sensor (like a CCD or CMOS sensor) which is a physical grid of millions of tiny, light-sensitive elements called **photodiodes**.
- **The Process:** Each sensor element "samples" a tiny portion of the continuous image. It measures the **average** intensity and color of all the light that falls on its specific grid square.

- **The Trade-off:** All the fine details *within* that single grid square are lost and averaged into a single value. The slides demonstrate this perfectly:
  - **High Sampling** (e.g., "256 lines") uses a fine grid with many small pixels, so it captures a lot of detail.
  - **Low Sampling** (e.g., "8 lines") uses a coarse grid with large pixels. Each pixel averages a huge area, resulting in a "blocky" or "pixelated" image where detail is lost.

3.3 Stage 3: Quantization (Continuous Value to Discrete Number)

This final stage takes the analog measurement from each pixel and discretizes its **value** (its brightness or color).

- **The Concept:** The continuous, analog signal (e.g., a voltage) measured by each pixel sensor is converted into a number from a **finite set of possible values**.
- **The Mechanism:** An Analog-to-Digital Converter (ADC) "rounds" the continuous measurement to the nearest allowed level.
- **The Process:** Most images use an 8-bit system, which provides **256 discrete levels** for each color (usually 0 for black and 255 for white). A sensor that measures a brightness of, say, 150.3 and another that measures 150.8 might both be "quantized" to the single digital value of 150.
- **The Trade-off:** Using too few levels (like 8, 4, or 2) results in visible "banding" or **posterization**, where smooth gradients are lost.

4 How Digital Sensors Capture Color

4.1 Anatomy of a Pixel

- **Microlens:** A tiny lens on each pixel to funnel in more light.
- **Color Filter:** A filter (Red, Green, or Blue) that only lets one color of light pass through.
- **Photodiode:** The component that converts light (photons) into an electrical signal (electrons).

4.2 The Bayer Filter and Demosaicing

- **Bayer Filter:** A mosaic of color filters on the sensor, typically 50% Green, 25% Red, and 25% Blue.
- **The Problem:** Each pixel only measures one color. A "Red" pixel has no idea what its Green or Blue value is.
- **Demosaicing:** The process where the camera's software "invents" the two missing colors for each pixel, often by interpolating from its neighbors.

5 Digital Image Representation

5.1 Image as a Matrix

- **Grayscale Image:** A 2D matrix where each element is a single number (e.g., 0-255) representing intensity.
- **Color Image:** A matrix of "triplets," where each element has three numbers, one for each color channel (e.g., R, G, B).

5.2 Color Spaces

- **RGB (Additive):** For light-emitting devices (monitors, projectors). Red + Green + Blue = White.
- **CMYK (Subtractive):** For printers. Uses Cyan, Magenta, Yellow, Black ink to subtract light from white paper.
- **YIQ (or YUV):** Used in broadcasting.
  - **Y:** Luminance (the B&W brightness component).
  - **I, Q:** Chrominance (the color information).

6 Point Operations

A function where a pixel's new value $g(x, y)$ depends **only** on its own original value $f(x, y)$.

$$g(x, y) = T(f(x, y)) \tag{2}$$

- **Examples:**
  - **Negative:** $T(u) = 255 - u$.
  - **Threshold:** $T(u) = \begin{cases} 0 & \text{if } u \leq 127 \\ 255 & \text{if } u > 127 \end{cases}$.
  - **Gamma (γ) Correction:** $T(u) = \text{Max} \cdot \left(\frac{u}{\text{Max}}\right)^{\gamma}$.
- **Look Up Table (LUT):** A very efficient method to perform point operations. An array is pre-calculated with the transformed value for all 256 possible inputs. The image is then transformed by "looking up" the new value for each pixel in this table.

7 Histograms

- A **histogram** is a bar chart showing the frequency (count) of each gray level (0-255) in an image.

- **Key Property:** A histogram is **invariant to pixel locations**. Different images can have the same histogram.
- Histograms can reveal an image's properties (e.g., dark, bright, or low-contrast).

7.1 Cumulative Histogram

- A "running total" of the histogram. The value $S(i)$ is the count of all pixels with a value of i or less.

$$S(i) = \sum_{j=0}^{i} h(j) \tag{3}$$

- **Application:** The distance between *cumulative* histograms ($|S1 - S2|$) is a much better metric for image similarity than the distance between regular histograms ($|h1 - h2|$). This is used for tasks like **video scene cut detection**.

8 Histogram Equalization

A powerful, automatic technique to improve image contrast by "spreading out" the histogram.

8.1 The Algorithm (Global Equalization)

1. Calculate the histogram $h(i)$ for the input image $I(x, y)$.
2. Calculate the cumulative histogram $S(i)$.
3. Find the minimum (m) and maximum (q) gray levels present in the image.
4. Create a Look Up Table (LUT) $T(i)$ that maps the old levels to a new, stretched range:

$$T(i) = \text{round}\left\{255 \times \frac{S(i) - S(m)}{S(q) - S(m)}\right\} \tag{4}$$

5. Apply the LUT to create the new, equalized image $J(x, y)$:

$$J(x, y) = T(I(x, y)) \tag{5}$$

Note: *This transformation is **monotonic**, meaning it does not reverse intensity order.*

8.2 Adaptive Histogram Equalization (AHE)

- **Problem:** Global equalization fails if an image has distinct regions (e.g., a bright area and a dark shadow).
- **Solution:** AHE is a *local* method. It computes a new equalization LUT for a small window around *each pixel*.

- **Result:** This brings out much more local detail but is computationally expensive and can amplify noise.

9 Next Topic

- The next lecture will cover the **Fourier Transform**.