# Overview of Computer Vision

Presented by

Srinivasan Thanukrishnan
CTO & Director
Glosys Technology Solutions Pvt. Ltd.
www.glosys.co.in

# Outline

1. **Computer Vision – An Overview**
2. **Image Formation & Representation**
3. **Imaging Geometry and Radiometry**
4. **Digitization in Computer Vision**
5. **Cameras and Projections**
6. **Rigid and Affine Transformation**
7. **Applications of Computer Vision**

# 1. Computer Vision – An Overview

- **Computer Vision** is a field of AI and computer science that enables machines to **see, interpret, and understand images or videos**, just like humans do.

- Fundamental Steps in Computer Vision

| Step | Description |
|---|---|
| **Image Acquisition** | Getting the image (camera, file, etc.) |
| **Preprocessing** | Cleaning, resizing, filtering |
| **Feature Extraction** | Edges, corners, textures, keypoints |
| **Detection/Recognition** | Object or pattern detection |
| **Post-processing** | Decision making, tracking, segmentation |

# 2. Image Formation & Representation

- A grayscale image is a **2D matrix** where each element represents **pixel intensity** (0 to 255).

- A color image is stored as **3 matrices**, one for each channel (Red, Green, Blue).

- Each pixel is represented as a triplet:
  Example at (0,0): (100,150,200)

- **Resolution**: number of pixels (e.g., 640×480)
- **Bit depth**: bits per pixel (e.g., 8-bit = 256 shades, 24-bit = RGB)

# Image Formation & Representation

- Image File Formats : bmp, png, jpeg

- Image Coordinate System: Origin usually at **top-left**. Coordinates increase **rightward (x)** and **downward (y)**.

- Each pixel is **0 or 1** (black or white). Useful in **thresholding** or **segmentation**

- Instead of RGB triplets, pixels store **index to color map**.

**Grayscale Conversion (RGB to Gray)**

- $Gray=0.2989 \cdot R+0.5870 \cdot G+0.1140 \cdot B$

# 3. Imaging Geometry

## Imaging Geometry

- Imaging geometry deals with **how 3D points in the world are projected into 2D points on an image plane** using a camera model.

## Camera Model – Pinhole Camera

- A simple model to explain image formation is the **pinhole camera model**.

## Geometry:

- 3D point: $P=(X,Y,Z)$; Image point (projected): $p=(x,y)$
- Focal length: $f$

## Projection Equation:

- $x=f \cdot XZ$,   $y=f \cdot YZ$

# Imaging Geometry

**Intrinsic Camera Parameters**

- Used to convert mm to pixel units on the image sensor.

**Intrinsic matrix K:**

- $K = \begin{bmatrix} fx & 0 & cx \\ 0 & fy & cy \\ 0 & 0 & 1 \end{bmatrix}$

Where

- fx, fy : focal lengths in pixel units
- cx, cy : image center (principal point)

Normalized point: (x,y) mm → convert to pixels:

- $u = fx \cdot x + cx$ $\qquad$ $v = fy \cdot y + cy$

# Imaging Geometry

**Extrinsic Camera Parameters**

- These describe the **position and orientation** of the camera in 3D space:

- $X_{camera} = R \cdot X_{world} + T$

Where:

- R: rotation matrix;
- T: translation vector

# Imaging Radiometry

- Radiometry studies how **light energy** interacts with surfaces and how it is **measured** by sensors.

**Basic Radiometric Quantities**

| Quantity | Symbol | Unit | Meaning |
|---|---|---|---|
| Radiant energy | ( Q ) | Joules (J) | Total energy |
| Radiant flux | ( \Phi ) | Watts (W = J/s) | Energy per second |
| Irradiance | ( E ) | ( W/m^2 ) | Power per unit area **on** a surface |
| Radiance | ( L ) | ( W/m^2/sr ) | Power per unit area per solid angle **from** a surface |

# Imaging Radiometry

**Irradiance ( E )**

- Light power hitting surface = $\Phi$
- Surface area = A
- $E=\Phi A$

**Radiance ( L )**

- Radiance is the **main quantity** that determines how bright a pixel appears.
- $L=d^2\Phi / ( dA\cdot\cos(\theta)\cdot d\omega )$

Where:

- dA: area on the surface
- $\theta$: angle between surface normal and light direction
- $d\omega$: solid angle subtended by the sensor

# Imaging Radiometry

**BRDF (Bidirectional Reflectance Distribution Function)**

- Describes how light reflects off a surface:
- $f_r(\theta_i,\theta_r)=dL_r / dE_i$
- Helps in realistic rendering & computer vision.

# Imaging Geometry Vs Radiometry

| Concept | Imaging Geometry | Radiometry |
|---|---|---|
| Core idea | Where does light land in the image | How much light is measured |
| Units | mm, pixels, angles | Watts, ( W/m^2 ), ( W/m^2/sr ) |
| Governs | Shape, position, perspective | Brightness, shading, exposure |
| Example | 3D point to 2D pixel | Irradiance = 5W / 2m² = 2.5 ( W/m^2 ) |

# 4. Digitization in Computer Vision

**Digitization (Sampling + Quantization)**

- Digitization is the process of converting an analog image (continuous in space and intensity) into a digital image (discrete in space and intensity).

**Spatial Sampling**

- Breaks down a continuous image into a **grid of pixels**.
- Each pixel represents a sample point in space.

Suppose we have a 1D intensity signal:

- Original analog signal (continuous):f(x) = [0, 20, 50, 100, 150, 200, 255] Now sample it every **2 points**:
- Sampled = [0, 50, 150, 255] → 4 pixels This is **spatial sampling**.

# Digitization in Computer Vision

## Quantization

- Converts continuous intensity values into **discrete levels** (e.g., 256 levels for 8-bit images).

- Let's say pixel intensity range is [0, 255] and we want to quantize to **4 levels**:

- Divide range: [0-63], [64–127], [128–191], [192–255]

- Original: [20, 150, 220]

- Quantized: [0, 2, 3]

# 5. Cameras

- Types of Cameras

| Camera Type | Description |
|---|---|
| Pinhole Camera | Simple model, no lens |
| Perspective Camera | Projects 3D → 2D with focal length |
| Orthographic Camera | Projects by ignoring depth (Z) |
| Digital Camera | Uses sensors (CCD or CMOS) to digitize images |

# Projections

**Perspective Projection**

- Objects farther from the camera appear smaller.
- Realistic, used in photography and vision.
- $x = f \cdot XZ$, $y = f \cdot YZ$
- Distant objects shrink under **perspective projection**

# Projections

**Orthographic Projection**

- Ignores depth Z, used in technical drawings.
- $x=X, \qquad y=Y$
- No perspective distortion.

# 6. Rigid & Affine Transformations

- In image processing, **transformations** are used to **map pixels** from one coordinate system to another — for alignment, registration, motion, warping, etc.

- Types of Transformations

| Transformation | Preserves | Changes Shape | Parameters |
|---|---|---|---|
| **Rigid** | Lengths, angles | No | Rotation + Translation |
| **Affine** | Parallelism, ratios | Yes (shear, scale) | Rotation + Translation + Scale + Shear |

# Rigid Transformations

- Rigid transformations preserve **distances and angles** (like moving or rotating a photo).

For a 2D point (x,y):

- $[x' \; y'] = R \cdot [x \; y] + [tx \; ty]$

Where:

- R is a **rotation matrix**

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

- tx, ty are **translations**

# Affine Transformations

**Affine Transformation**

Affine transformation is **more general**. It includes:

- Translation
- Rotation
- Scaling
- Shearing

- It **preserves straight lines and parallelism**, but not distances or angles.

# 7. Applications of Computer Vision

**Image Classification**

- Predicting object class in image

**Object Detection**

- Locate objects using bounding boxes

**Semantic Segmentation**

- Label each pixel as part of a class
  Road, car, pedestrian...

**Facial Recognition**

- Matching faces using features
  Image → face embedding → match

# Applications of Computer Vision

**Medical Imaging**

- Detect tumors in MRI
- Segment organs

**Autonomous Driving**

- Detect roads, signs, lanes, and people

**Agriculture**

- Identify plant diseases from images