Day 2: Understanding Video Streams in OpenCV

Outcomes:

- Understand how frames are read from a camera
- Know what `ret` and `frame` mean
- Understand how OpenCV creates live video using frames
- Clearly know why `while True` is mandatory

Concept of a Video

- Video ≠ Video
- Camera does not give a video file
- It gives a stream of images one at a time called as Frames

Note: A video stream in OpenCV is created by continuously reading and displaying individual frames inside a loop.

What is ret and frame?

```
ret, frame = cap.read()
```

Core concept:

- `cap` is an object of class `cv2.VideoCapture(0)`
- `read()` is a method of the class `cv2.VideoCapture(0)`

How does the `read()` method work?

- The `read()` method returns two values: `ret` and `frame`
- `ret` type: boolean
- `frame` type: NumPy array
- Consider the code:

```
if not ret:
    break
```

- it checks if the frame is read successfully?
    - if True: frame exists
    - if False: camera failed / video ended

Note: The `read()` method reads a single frame from the video source and returns whether the read was successful along with the frame data.

Why do we need a `while` loop?

Reading and displaying a frame only shows a frozen image, while we need a live video output.

Hence, we need to continuously

1. read frames,
2. store them and
3. display them one after another.

Producing a live video output.

This can be achieved by using a loop, as a **video is just a stream of frames.**

Structure of the loop:

```
while True:
    read frame
    show frame
```