

Lab 1: Introduction to OpenCV & Its Importance

📅 **Lab Duration:** 3 Hours

📌 **Objective:**

- Understand **what OpenCV is** and **why it is important** in modern applications.
 - Install OpenCV and set up the development environment.
 - Learn basic **image operations** (reading, displaying, saving images).
 - Explore **real-world applications of OpenCV**.
-

📁 Lab Outline

**1. What is OpenCV? **

- OpenCV (Open Source Computer Vision Library) is a popular open-source library for **image processing, computer vision, and machine learning**.
- It is used for applications such as **face recognition, object tracking, medical imaging, and self-driving cars**.

📌 Why Use OpenCV?

- ☑ Fast and efficient image processing.
 - ☑ Supports multiple programming languages (Python, C++, Java).
 - ☑ Cross-platform (Windows, Linux, macOS, Android).
 - ☑ Open-source and widely adopted in **AI & machine learning**.
-

2. Installing OpenCV (30 mins)

Step 1: Install OpenCV

- Open **Command Prompt (Windows)** or **Terminal (Linux/Mac)** and run:
`pip install opencv-python`
- If you are using OpenCV without GUI support, install:
`pip install opencv-python-headless`

Step 2: Verify Installation

Run the following Python script to check the installation:

```
import cv2
print("OpenCV Version:", cv2.__version__)
```


If OpenCV is correctly installed, it will print the version number.

3. Reading, Displaying, and Saving Images (60 mins)

🚀 *Read an Image*

```
import cv2

# Read an image from file
image = cv2.imread("sample.jpg")

# Check if the image is loaded
if image is None:
    print("Error: Could not read image")
else:
    print("Image loaded successfully!")
```

🚀 *Display an Image*

```
cv2.imshow("Sample Image", image) # Show image in a window
cv2.waitKey(0) # Wait for a key press to close the window
cv2.destroyAllWindows() # Close all windows
```

🚀 *Convert Image to Grayscale*

```
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
cv2.imshow("Grayscale Image", gray_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

🚀 *Save the Grayscale Image*

```
cv2.imwrite("grayscale_sample.jpg", gray_image) # Save the image
print("Image saved successfully!")
```

4. Real-World Applications of OpenCV

- **Face Recognition** – Used in biometric authentication (Face ID).
 - **Self-Driving Cars** – Object detection & lane tracking.
 - **Medical Imaging** – Detect tumors in X-rays and MRI scans.
 - **Security & Surveillance** – Motion detection in CCTV cameras.
 - **Retail & Augmented Reality** – Virtual try-ons (e.g., Snapchat filters).
-

****5. Hands-on Exercises ****

✂ **Task 1:** Load and display a new image of your choice.

✂ **Task 2:** Convert the image to grayscale and save it with a new filename.

✂ **Task 3:** Research and write a **one-page report** on an application of OpenCV.

□ Expected Outcome

By the end of this lab, students should be able to:

- ☒ Understand **why OpenCV is important**.
- ☒ Install OpenCV and verify its installation.
- ☒ Perform **basic image operations** (reading, displaying, converting, saving).
- ☒ Recognize **real-world applications** of OpenCV.