S. B. JAIN INSTITUTE OF TECHNOLOGY, MANAGEMENT & RESEARCH, NAGPUR.

Practical No. 12

PostLab

Aim: Develop a program to implement the Viola-Jones algorithm for detecting faces from video.

Name of Student: Shrutika Pradeep Bagdi

Roll No.: CS22130

Semester/Year: VII/IV

Academic Session: 2025-2026

Date of Performance:

Date of Submission:

AIM: Develop a program to implement the Viola-Jones algorithm for detecting faces from video.

OBJECTIVE:

- Understand the Viola-Jones Framework: To grasp the fundamental principles of the algorithm, including the role of Haar-like Features, the Integral Image representation, AdaBoost learning, and the Cascade of Classifiers structure.

- Apply Pre-trained Models: To practically utilize pre-trained Haar Cascade XML files (an output of the Viola-Jones training process) for rapid, real-time object detection.

- Video Processing: To master the basics of capturing video input, processing it frame-by-frame, and drawing bounding boxes to localize the detected object (face).

- Real-Time Performance: To observe the high speed and efficiency of the cascaded classifier architecture, which makes real-time video processing possible.

THEORY:

The Viola-Jones algorithm, pioneered by Paul Viola and Michael Jones in 2001, revolutionized the field of face detection. Its efficient and robust methodology opened doors to a wide range of applications that rely on accurately identifying and analyzing human faces. By harnessing the power of Haar-like features, integral images, machine learning, and cascades of classifiers, the Viola-Jones algorithm showcases the synergy between computer science and image processing.

The Viola-Jones algorithm is a landmark object detection method known for achieving high detection rates with minimal computational cost, enabling real-time operation. It consists of four key concepts:

The Viola Jones algorithm has four main steps, which we shall discuss in the sections to follow:

1. Selecting Haar-like features
2. Creating an integral image
3. Running AdaBoost training
4. Creating classifier cascades

1. Haar-like Features

These are simple rectangular features that capture contrast changes in an image. They are analogous to Haar wavelets and are used because human faces exhibit universal properties, such as:

- The **eye region** is darker than the **cheek region**.
- The **bridge of the nose** is brighter than the **eye regions**. A Haar-like feature calculates a value by taking the sum of pixels in the dark areas and subtracting the sum of pixels in the light areas.

2. Integral Image

This is a pre-computation step that allows the sum of pixels within any rectangular region to be calculated in constant time (only four array accesses), regardless of the rectangle's size. This dramatically speeds up the calculation of the thousands of possible Haar-like features across an image.

3. AdaBoost (Adaptive Boosting)

AdaBoost is a machine learning algorithm used to select a small, critical set of **"strong classifiers"** from a very large pool of weak classifiers (each corresponding to a single Haar-like feature). It focuses the learning process on the most challenging examples, creating a highly accurate combined classifier using only the most important features.

4. Cascade of Classifiers

The final, most critical component for speed. The strong classifier is organized into a cascade (a series of stages).

- **Early Stages** use a small number of simple features to quickly eliminate most non-face regions.
- **Later Stages** use more complex features and are only applied to regions that passed the previous stages. This structure ensures that the majority of an image's sub-regions are instantly discarded as "non-face," allowing the system to achieve processing speeds necessary for real-time video analysis.

ALGORITHM:

Step 1: Initialization

Step 2. Video Input

Step 3. Frame Loop

Step 4. Pre-processing

Step 5. Detection

Step 6. Output Processing

Step 7. Drawing/Display

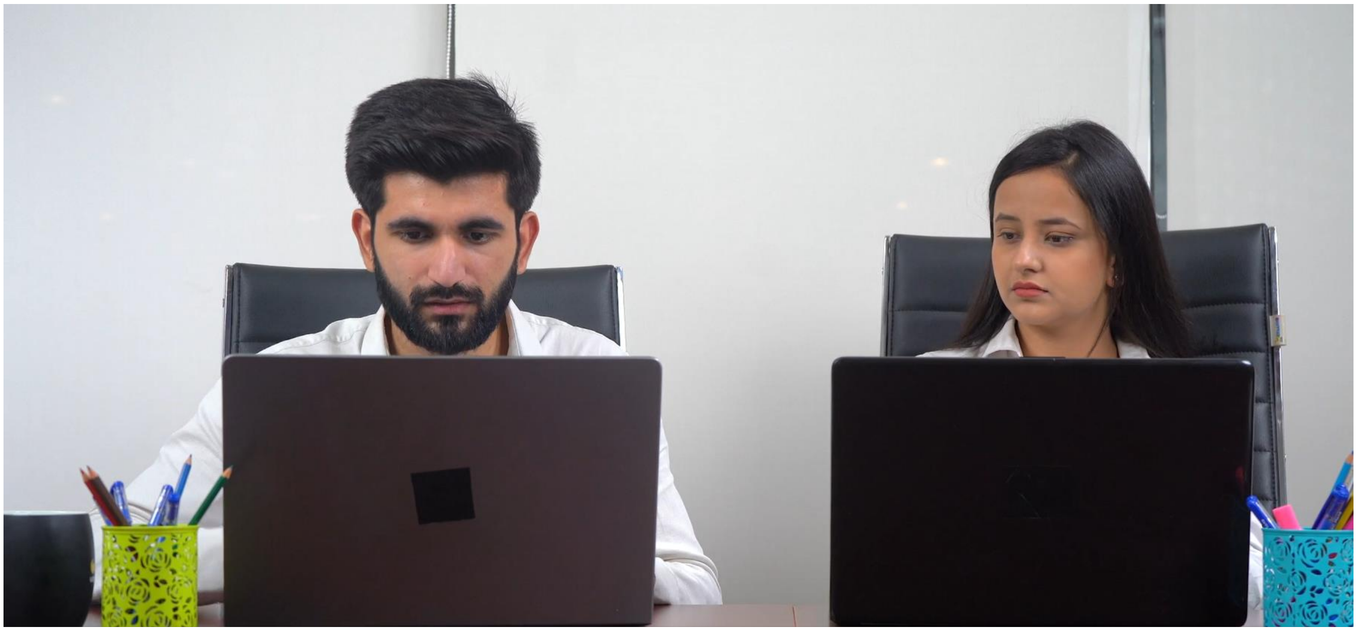
Step 8. Display Frame

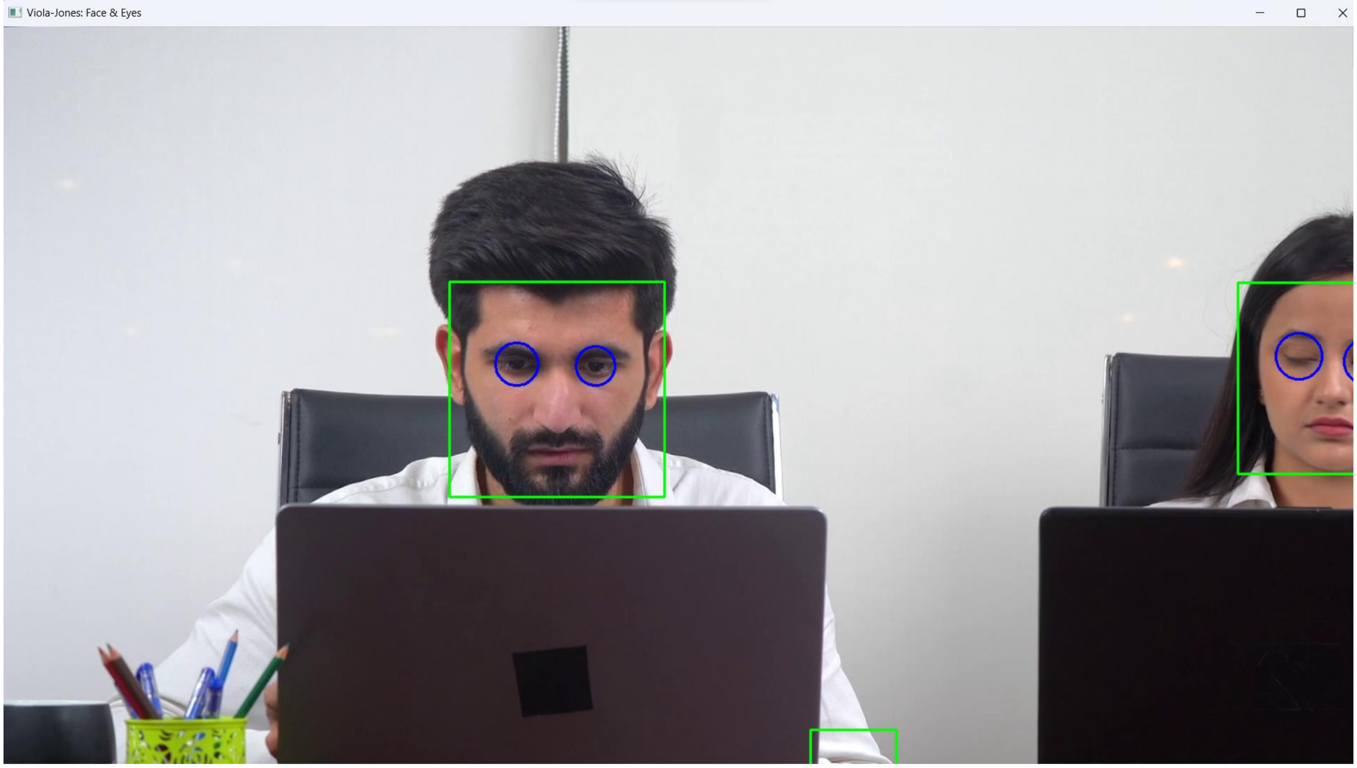
Step 9. Termination

Step 10. Cleanup

CODE:

Input and Output image:





CONCLUSION: The Viola-Jones algorithm was successfully implemented in real-time video using Python and OpenCV's Haar Cascade implementation.

DISCUSSION AND VIVA VOCE:

1. What are the four key components of the Viola-Jones algorithm?
2. What is an Integral Image, and why is it used?
3. What is the role of AdaBoost in this algorithm?
4. Why is the Cascade Structure important for real-time detection?

REFERENCE:

- https://www.mygreatlearning.com/blog/viola-jones-algorithm/
- https://www.geeksforgeeks.org/computer-vision/a-complete-guide-to-face-detection-and-face-recognition-in-2024/
- https://realpython.com/traditional-face-detection-python/
- https://www.datacamp.com/tutorial/face-detection-python-opencv