**Live Face Detection in Video Stream**

**1. Objective: -**

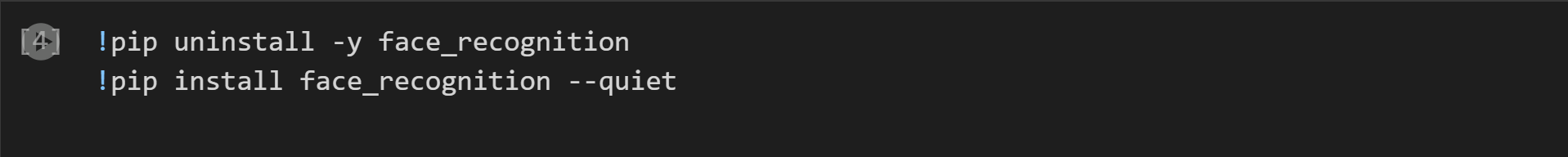
To build a simple Python-based application that detects human faces in a video using the face recognition API.

This project is designed for beginners in coding and computer vision to learn how to process video, detect faces, and visualize results — all inside Google Colab .

**2. Tools & Technologies: -**

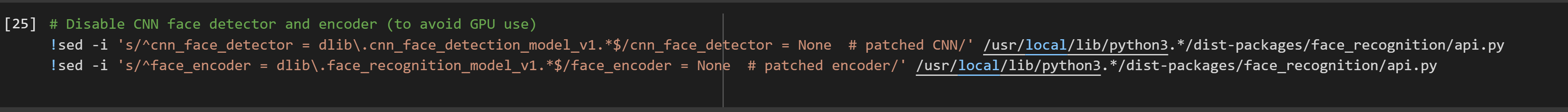
* Google Colab
* Python 3
* OpenCV (cv2)
* face\_recognition
* NumPy
* IPython.display

**Step 1: Install the Required Library**



* What it does:
  + This installs the face\_recognition Python package, which helps us detect and recognize faces in pictures and videos.
* Why it's important:
  + Without installing this library, we can't use the tools to detect faces. The first line removes the old version just in case, and the second line installs a fresh one.

**Step 2: Modify Settings to Avoid Using GPU**



* What it does:
  + This code changes settings inside the face recognition library so it doesn’t try to use a special processor (GPU) that Google Colab doesn't always support.
* Why it's important:
  + If we don’t do this, the code might crash or give errors. This change makes it work more smoothly in Colab.

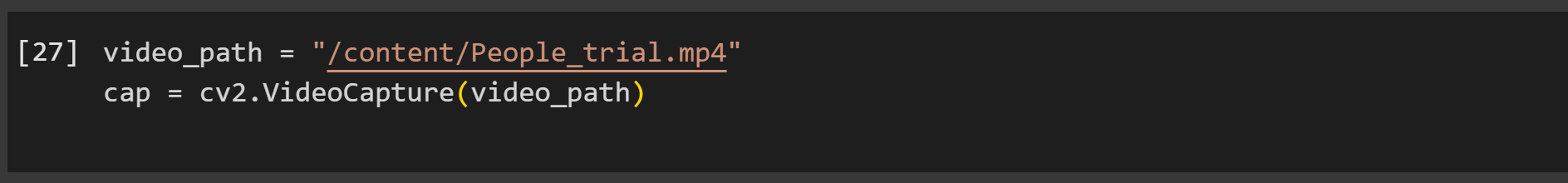
**Step 3: Import the Required Libraries**

A screenshot of a computer

AI-generated content may be incorrect.

* Press Shift + Enter or click the play/run ▶️ button.

**Step 4: Load the Video File**



* This loads a video file (People\_trial.mp4) so we can analyze it frame by frame.
* We need to load the video into a format that Python (through OpenCV) can work with to find faces in different frames.

**Step 5: Choose Random Frames to Analyze**

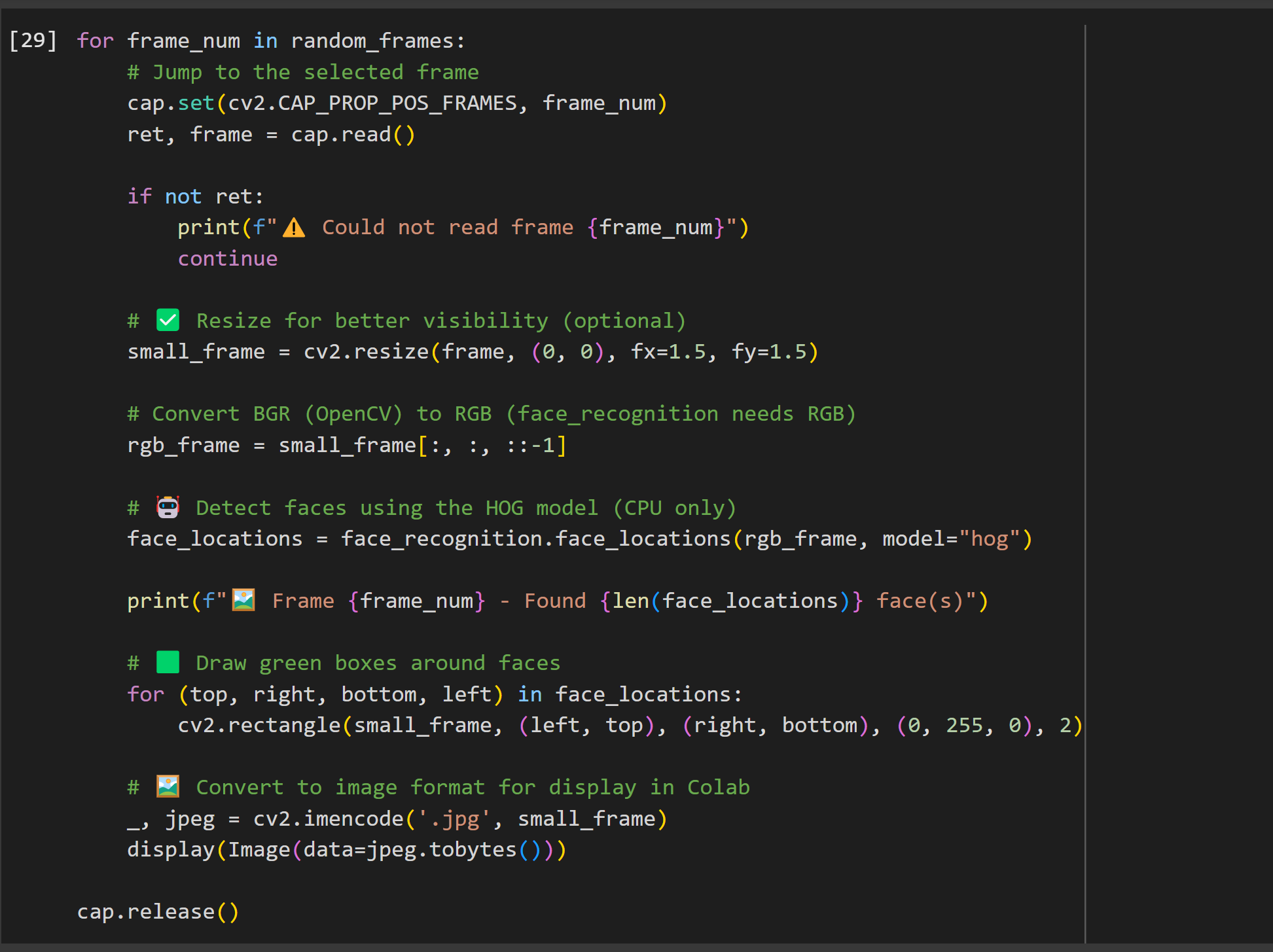
A screen shot of a computer code

AI-generated content may be incorrect.

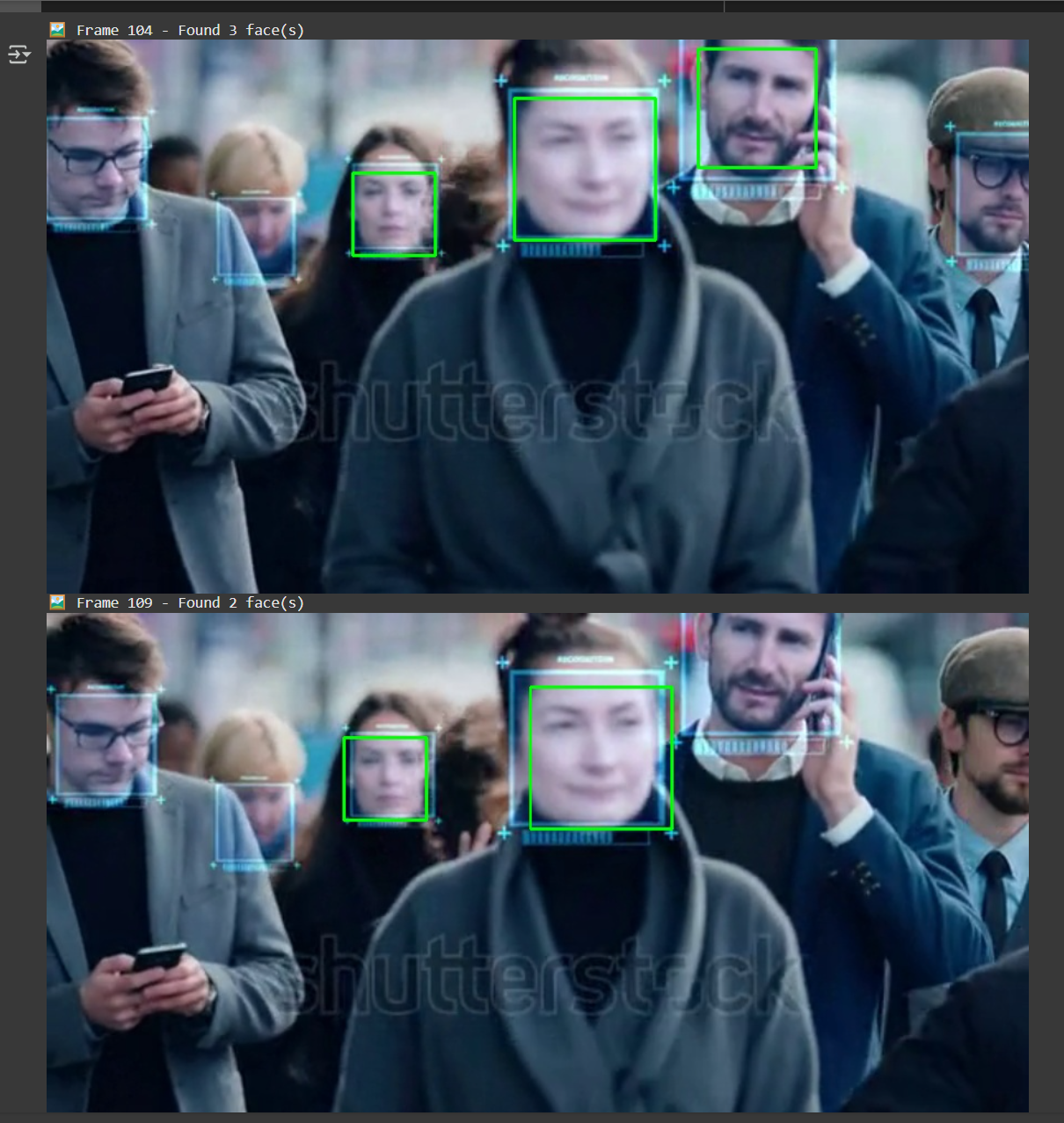
* Counts how many frames are in the video.
* Picks 5 random frames to analyze.
* Checking every frame can be slow, so we pick a few to test how well face detection works.
* Click the Run ▶️ button or press Shift + Enter.
* You will see:



**Step 6: Detect Faces and Display the Results**

****

* Jumps to each chosen frame.
* Detects faces using the HOG model (CPU-based).
* Draws green rectangles around faces.
* Shows the frame in the notebook.
* This is the main logic that checks for faces and shows results visually, which is key to seeing if our detection works.
* Click the Run ▶️ button or press Shift + Enter.
* You will see:



* For each selected frame, an image is shown with green rectangles drawn around any detected faces.
* If no faces are found, the image will still be shown but with no green boxes.

A screenshot of a computer

AI-generated content may be incorrect.

* If you don’t see any faces detected, that doesn’t always mean the code is broken. It could mean:
* The face is turned away.
* The image is blurry.
* The lighting is poor.
* Or the frame didn’t load properly.