# Import libraries

import cv2

import numpy as np

import dlib

Step 2: Open the default camera to capture faces and use the dlib library to

get coordinates.

• Python3

# (0) in VideoCapture is used to

# connect to your computer's default camera

cap = cv2.VideoCapture(0)

# Get the coordinates

detector = dlib.get\_frontal\_face\_detector()

Step 3: Count the number of faces.

• Capture the frames continuously.

• Convert the frames to grayscale(not necessary).

• Take an iterator i and initialize it to zero.

• Each time you get the coordinates to the face structure in the

frame, increment the iterator by 1.

• Plot the box around each detected face along with its face count.

• Python3

while True:

 # Capture frame-by-frame

 ret, frame = cap.read()

 frame = cv2.flip(frame, 1)

 # Our operations on the frame come here

 gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

 faces = detector(gray)

 # Counter to count number of faces

 i = 0

 for face in faces:

 x, y = face.left(), face.top()

 x1, y1 = face.right(), face.bottom()

 cv2.rectangle(frame, (x, y), (x1, y1), (0, 255, 0), 2)

 # Increment the iterartor each time you get the coordinates

 i = i+1

 # Adding face number to the box detecting faces

 cv2.putText(frame, 'face num'+str(i), (x-10, y-10),

 cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (0, 0, 255), 2)

 print(face, i)

 # Display the resulting frame

 cv2.imshow('frame', frame)

Step 4: Terminate the loop.

• Python3

# Enter key 'q' to break the loop

if cv2.waitKey(1) & 0xFF == ord('q'):

 break

Step 5: Clear windows.

• Python3

# When everything done, release

# the capture and destroy the windows

cap.release()

cv2.destroyAllWindows()

Below is the complete program of the above approach:

• Python3

# Import required libraries

import cv2

import numpy as np

import dlib

# Connects to your computer's default camera

cap = cv2.VideoCapture(0)

# Detect the coordinates

detector = dlib.get\_frontal\_face\_detector()

# Capture frames continuously

while True:

 # Capture frame-by-frame

 ret, frame = cap.read()

 frame = cv2.flip(frame, 1)

 # RGB to grayscale

 gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

 faces = detector(gray)

 # Iterator to count faces

 i = 0

 for face in faces:

 # Get the coordinates of faces

 x, y = face.left(), face.top()

 x1, y1 = face.right(), face.bottom()

 cv2.rectangle(frame, (x, y), (x1, y1), (0, 255, 0), 2)

 # Increment iterator for each face in faces

 i = i+1

 # Display the box and faces

 cv2.putText(frame, 'face num'+str(i), (x-10, y-10),

 cv2.FONT\_HERSHEY\_SIMPLEX, 0.7, (0, 0, 255), 2)

 print(face, i)

 # Display the resulting frame

 cv2.imshow('frame', frame)

 # This command let's us quit with the "q" button on a keyboard.

 if cv2.waitKey(1) & 0xFF == ord('q'):

 break

# Release the capture and destroy the windows

cap.release()

cv2.destroyAllWindows()