TEAM NAME :- FAUX MECHANISM CREW

VELAMMAL COLLEGE OF ENGINEERING AND TECHNOLOGY

JEYAVANI SANDHYA .C

VANDHANA R.B

NIVEDTHITHA .T

|  |  |
| --- | --- |
| Related image  machine learning  **Student profile** | Abstract  Problem statement and source code for student profiling using machine learning algorithm.(supervised algorithm) |

**PROBLEM STATEMENT:**

* **Open CV- Face detection**
* **Python 3- juypternotebook**
* **Data set –image processing**

The student profiling system involves recognizing the photo of a student and display the profile of the student .The student data set is to be created first. The data set is created by capturing the photos of the students with minute details. The relevant details for the student should be provided already. When the student face is identified by the system it checks for the data set whether the profile is present or not. It compares the photo with the data set. If the photo is found in the dataset then the particular student profile will be shown .If the photo is not found then the profile of the student could be added to the dataset.

* Display the name of a student when face identified according to the dataset
* Predect the future marks with another data set.

**The faces of the students are already created ie.( Labeled. ). When a face in infront of the Open CV . The profile is displayed.**

**LOGIC OF THE PROGRAM:**

import cv2

face\_cascade = cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

eye\_cascade = cv2.CascadeClassifier('haarcascade\_eye.xml')

cap = cv2.VideoCapture(0)

while 1:

# reads frames from a camera

ret, img = cap.read()

# convert to gray scale of each frames

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

# Detects faces of different sizes in the input image

faces = face\_cascade.detectMultiScale(gray, 1.3, 5)

for (x,y,w,h) in faces:

# To draw a rectangle in a face

cv2.rectangle(img,(x,y),(x+w,y+h),(255,255,0),2)

roi\_gray = gray[y:y+h, x:x+w]

roi\_color = img[y:y+h, x:x+w]

# Detects eyes of different sizes in the input image

eyes = eye\_cascade.detectMultiScale(roi\_gray)

#To draw a rectangle in eyes

for (ex,ey,ew,eh) in eyes:

cv2.rectangle(roi\_color,(ex,ey),(ex+ew,ey+eh),(0,127,255),2)

# Display an image in a window

cv2.imshow('img',img)

# Wait for Esc key to stop

k = cv2.waitKey(30) & 0xff

if k == 27:

break

# Close the window

cap.release()