import face\_recognition

import cv2

import numpy as np

import csv

import os

from datetime import datetime

video\_capture=cv2.VideoCapture(0)

jobs\_image=face\_recognition.load\_image\_file("photos/jobs.jpg")

jobs.encoding=face\_recognition.face\_encoding(jobs\_image)[0]

siddaramaiah\_image=face\_recognition.load\_image\_file("photos/siddaramaiah.jpg")

siddaramaiah\_encoding=face\_recognition.face\_encoding(siddaramaiah\_image)[0]

yediyurappa\_image=face\_recognition.load\_image\_file("photos/yediyurappa.jpg")

yediyurappa\_encoding=face\_recognition.face\_encoding(yediyurappa\_image)[0]

kumarswami\_image=face\_recognition.load\_image\_file("photos/kumarswami.jpg")

kumarswami\_encoding=face\_recognition.face\_encoding(kumarswami\_image)[0]

known\_face\_encoding=[jobs\_encoding,siddaramaiah\_encoding,yediyurappa\_encoding,kumarswami\_encoding]

students=known\_face\_names.copy()

face\_location=[]

face\_encoding=[]

face\_names=[]

s=True

now=datetime.now()

current\_date=now.strftime("%y-%m-%d")

f=open('current\_date+.csv','w+',newline='')

lnwriter=csv.writer(f)

while True:

\_,frame=video\_capture,read()

small\_frame=cv2.resize(frame,(0,0),fx=0.24,fy=0.25)

rgb\_small\_frame=smallsmall\_frame[:,:,::,-1]

if s:

face\_locations=face\_recognition.face\_locations(rgb\_small\_frame)

face\_encodings=face\_recognition.face\_encodings(rgb\_small\_frame,face\_locations)

face\_name=[]

for face\_encoding in face\_encodings:

matches=face\_recognition.compare\_faces(known\_face\_encoding,face\_encoding)

name=""

face\_distance=face\_recognition.face\_distance(known\_face\_encoding)

best\_match\_index=np.argmin(face\_distance)

if matches[best\_match\_index]:

name=known\_faces\_names[best\_match\_index]

face\_names.append(name)

if name in known\_faces\_names:

if name in students:

students.remove(name)

print(students)

current\_time=now.strftime("%H-%M-%S")

lnwriter.writerow([name,current\_time])

cv2.imshow("attendence system",frame)

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

break

video\_capture.release()

cv2.destroyA11Windows()

f.close()