COMPUTER SCIENCE AND ENGINEERING

Face recognition using open cv and python

Artificial Intelligence

Course code : CSE 418

Submitted to:

Supta Richard Philip
Lecturer of City University
Dept. of CSE

Submitted by:

Student Name:	Student Id:
Nusrat Jahan Muna	161412314
Rabeya khatun	161412333
Md Al Amin Pramanik	161412344
Nusrat Jahan	161412336

Abstract

Identifying a person with an image has been popularised through the mass media. However, it is less robust to fingerprint or retina scanning. This report describes the face detection and recognition mini-project undertaken for the visual perception and autonomy module at Plymouth university. It reports the technologies available in the Open-Computer-Vision (OpenCV) library and methodology to implement them using Python. For face detection, Haar-Cascades were used and for face recognition Eigenfaces, Fisherfaces and Local binary pattern histograms were used. The methodology is described including flow charts for each stage of the system. Next, the results are shown including plots and screen-shots followed by a discussion of encountered challenges. The report is concluded with the authors' opinion on the project and possible applications.

Contents

Chapter 1

Introduction

1.1 Problem Statement

Human facial expressions can be easily classified into 7 basic emotions: happy, sad, surprise, fear, anger, disgust, and neutral. Our facial emotions are expressed through activation of specific sets of facial muscles. These sometimes subtle, yet complex, signals in an expression often contain an abundant amount of information about our state of mind. Over the past decade face detection and recognition have transcended from esoteric to popular areas of research in computer vision and one of the better and successful applications of image analysis and algorithm based understanding. Because of the intrinsic nature of the problem, computer vision is not only a computer science area of research, but also the object of neuro-scientific and psychological studies also, mainly because of the general opinion that advances in computer image processing and understanding research will provide insights into how our brain work and vice versa.A general statement of the face recognition problem can be formulated as follows: given still or video images of a scene, identify or verify one or more persons in the scene.imageprocesssing using opencv python or face recognition using opencv python.

1.2 Objectives

• Provides a valuable attendance service for both teachers and students.
• Reduce manual process errors by provide automated and a reliable attendance system uses face recognition technology.
• Increase privacy and security which student cannot presenting himself or his friend while they are not.
• Produce monthly reports for lecturers.
• Flexibility, Lectures capability of editing attendance records.

Chapter 2

Background Study

2.1 Related Work 1

We read an article based on face detection.That paper brings together new algorithms and insights to construct a framework for robust and extremely rapid visual detection. There are three main contributions of our face detection framework. We will introduce each of these ideas briefly below and then describe them in detail in subsequent sections. The first contribution of this paper is a new image representation called an integral image that allows for very fast feature evaluation. The second contribution of this paper is a simple and efficient classifier that is built by selecting a small number of important features from a huge library of potential features using AdaBoost. The third major contribution of this paper is a method for combining successively more complex classifiers in a cascade structure which dramatically increases the speed of the detector by focusing attention on promising regions of the image.

2.2 Related Work 2

We read another an article based on face detection.They Proposed a face detection algorithm for colour image in the presence varying lighting condition as well as complex backgrounds.Based on a novel lighting compensation technique ,their method detects skin regions over entire image .Then algorithm constract eye ,mouth and boundary maps for verifying each other candidates .Experiment result demonstrate successful face detection over a wide range of facial variation in colour ,position ,scale and expression in image from several photos.

Chapter 3

Methodology

3.1 What is Requirement

In product development and process optimization, a requirement is a singular documented physical or functional need that a particular design, product or process aims to satisfy.Needed requirement 1.Python
2.Open cv
3.Numpy
4.Mathplotlib

3.2 Working Process

3.2.1 Install open cv pyhon

Need both main and contrib modules (check extra modules listing from OpenCV documentation) OpenCV-Python supports all the leading platforms like Mac OS, Linux, and Windows. It can be installed in either of the following ways: Unofficial pre-built OpenCV packages for Python. Packages for standard desktop environments (Windows, macOS, almost any GNU/Linux distribution) run pip install opencv-python if we need only main modules run pip install opencv-contrib-python if we can either use Jupyter notebooks or any Python IDE of choice for writing the scripts.Download OpenCV from

Sourceforge. Go to OpenCV/build/python/2.7 folder. Copy cv2.pyd

3.2.2 Import the necessary libraries

import numpy as np import cv2 import matplotlib.pyplot as plt Read in the image using the imread function. We will be using the colored 'mandrill' image for demonstration purpose. It can be downloaded from here img$_{r}aw = cv2.imread('image.jpg')$

3.2.3 Test

Before using function image is not detected .we are check in project several times.It only detect when the face come to the cam.

3.2.4 Face detection

We shall be using the detectMultiscale module of the classifier. This function will return a rectangle with coordinates(x,y,w,h) around the detected face. This function has two important parameters which have to be tuned according to the data. scalefactor In a group photo, there may be some faces which are near the camera than others. Naturally, such faces would appear more prominent than the ones behind. This factor compensates for that. minNeighbors This parameter specifies the number of neighbors a rectangle should have to be called a face. You can read more about it here. faces$_{r}ects = haar_{c}ascade_{f}ace.detectMultiScale(test_{i}mage_{g}ray,$ $scaleFactor = 1.2, minNeighbors = 5);$

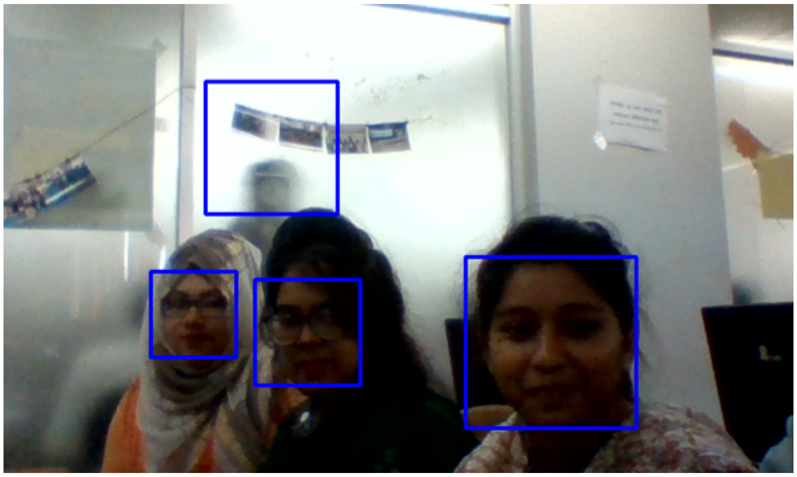
Let us print the no. of faces found print('Faces found: ', len(faces$_{r}ects$))

Faces found: 1 Our next step is to loop over all the coordinates it returned and draw rectangles around them using Open CV. We will be drawing a green rectangle with a thickness of
for (x,y,w,h) in faces$_{r}ects$:
$cv2.rectangle(test_{i}mage,$
$(x, y), (x + w, y + h), (0, 255, 0), 2)$
$test_{i}mage2 = cv2.imread('group.jpg')$

call the function to detect faces
faces = detect$_{f}aces(haar_{c}ascade_{f}ace, test_{i}mage2)$

convert to RGB and display image
plt.imshow(convertToRGB(faces))

Finally, we shall display the orginal image.
convert image to RGB and show image



Here, it is. We have successfully detected in the picture.

Chapter 4

Evaluation

4.1 Error in the model

1.Our project cannot detect face in low light . 2.Cannot show face detect person name.

4.2 Future work

we will solve all problem in future .we will solve that the face accurately detect in low light and show the detect person name .

Chapter 5

Conclusion

We have presented an approach for face detection which minimizes computation time while achieving high detection accuracy. The approach was used to construct a face detection system which is approximately 15 times faster than any previous approach. Preliminary experiments, which will be described elsewhere, show that highly efficient detectors for other objects, such as pedestrians or automobiles, can also be constructed in way.

5.1 References

1.Viola, P. and Jones, M.J., 2004. Robust real-time face detection. International journal of computer vision, 57(2), pp.137-154.
2.Hsu, R.L., Abdel-Mottaleb, M. and Jain, A.K., 2002. Face detection in color images. IEEE transactions on pattern analysis and machine intelligence, 24(5), pp.696-706.