FaceFilter App

Mitin Sharma

November 2017

1 Abstract

Face detection is a computer technology being used in a variety of applications that identifies human faces in digital images. Face detection also refers to the psychological process by which humans locate and attend to faces in a visual scene. Face detection can be regarded as a specific case of object-class detection. In object-class detection, the task is to find the locations and sizes of all objects in an image that belong to a given class. Examples include upper torsos, pedestrians, and cars. Face-detection algorithms focus on the detection of frontal human faces. It is analogous to image detection in which the image of a person is matched bit by bit. Image matches with the image stores in database. Any facial feature changes in the database will invalidate the matching process. A reliable face-detection approach based on the genetic algorithm and the eigenface technique. Firstly, the possible human eye regions are detected by testing all the valley regions in the gray-level image. Then the genetic algorithm is used to generate all the possible face regions which include the eyebrows, the iris, the nostril and the mouth corners.

2 Introduction

In today’s era, we have many apps available on e-stores that detect face and other parts. The main objective of my application is to learn machine learning concepts and implement my own logics to detect eyes and face on a human and put a filter on the face of human. For example: Snapchat filters, facebook filters. I have used OpenCV and Python to develop this app. This App take input as an image from webcam and right away replace face with the filters. Currently i made few filters for my app that are Hat, Moustaches, Dog face, Cowboy hat.

3 Related Work

Initially, i worked on tensor flow technology to achieve this goal. Later i found, using OpenCV is the best python library to implement my logic.

4 Methodology

To construct this application we are making use of OS resources like console, webcam drivers as plan on building a runtime app.

5 Experiments

We have tried to implement logic in tensor flow and other available python libraries.

6 Analysis

With some minor tweaks to the code, everything worked pretty well, and this has further given us a confidence towards implementing FaceFilter App.

7 Conclusion

Experiment with the Tensor Flow and OpenCV library were exciting part that we had done so far. We are looking forward to implement FaceFilter app using OpenCV and make our app robust; so that face and eye detection can be done to replace with funny filters. This certainly seems an uphill task given the time constraint but we are pretty sure we could make this happen.

8 References

- https://www.python.org/doc/
- http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html
- http://www.numpy.org/