

Python face-recognition Library Introduction and Overview

Python face-recognition library is a simple, user-friendly library with methods useful for to recognize and manipulate faces from Python. More details can be found from www.pypi.org/project/face-recognition/.

face-recognition 1.3.0

pip install face-recognition

Latest version

Released: Feb 20, 2020

Recognize faces from Python or from the command line

Navigation

- Project description
- Release history
- Download files

Project description

Face Recognition

Recognize and manipulate faces from Python or from the command line with the world's simplest face recognition library.

Built using [dlib](#)'s state-of-the-art face recognition built with deep learning. The model has an accuracy of 99.38% on the [Labeled Faces in the Wild](#) benchmark.

This also provides a simple `face_recognition` command line tool that lets you do face recognition on a folder of images from the command line!

ppi v1.3.0

build passing

docs passing

Python library **Face-recognition** is built on three important foundations.

1. CMake
2. Dlib
3. Open CV



CMake: CMake is a cross-platform free an open source software tool. This is used to manage the

software building process using compiler independent method.

1.



Dlib is a dynamic library. This is actually a modern C++ to solve real life problem. This contains machine learning algorithms and tools for building complex software in C++ to solve real life problem. Most of the Machine Learning packages are built on Dlib

2.



Open CV (opensource computer vision)

This is a very popular opensource library implementing Computer Vision algorithms using Machine Learning.

Python Face Recognition



Operating System

Major Features

This library has the following key features:

1. Face detection and location

Copyright Dr. Xavier Chelladurai 17 July 2021

Face detection is about detecting human faces in a picture. This identifies all the human faces in the picture and locate them in the 2-dimensional representation as rectangle frames. This can be done using two different methods.

1. **Histogram Oriented Gradient** approach denoted as hog approach.
2. Deep Learning based Convolution Neural Network (CNN) based approach

While hog approach is simple and runs in less computing time its accuracy is low. CNN is very complex and takes more time but has a better accuracy level.

Model	Speed	Accuracy
HOG	Fast	Low
CNN	Slow	High

2. Face Encoding

Face Encoding is about extracting the features of faces and convert them into their respective face signatures. The signature is further used for comparing faces and recognize a resemblance.

3. Face Resemblance Identification

Given two faces, it is possible to check the resemblance. This is done as follows:

1. Load face f_1 and f_2
2. Find Face encoding of f_1 and f_2 and get their signatures s_1 and s_2 .
3. Find the distance between s_1 and s_2 distance (s_1, s_2)
4. If distance (s_1, s_2) is small we say the face f_1 and f_2 have resemblance.

4. Face Landmark Identification

It can identify important landmarks on human faces. The following are identified:

1. Chin
2. Left eyebrow
3. Right eyebrow
4. Nose bridge
5. Nose tip
6. Left eye
7. Right eye
8. Top lip
9. Bottom lip.

5. Face recognition

There are some known faces that are registered in the memory. When a new face f_u is received we can recognize this new face f_u as one of the known faces. This feature is used to be used in the face-recognition-based Employee Attendance System and Immigration Systems.