

Documentación de código “reconocimiento de rostros”

Python



15 de marzo de 2017

Unidad de soporte De la facultad de matemáticas

José Omar León Peraza, Israel Fuentes Brito

Contenido

[Introducción 2](#_Toc487126544)

[Objetivo del código 2](#_Toc487126545)

[Archivos 2](#_Toc487126546)

[Face\_detect.py 2](#_Toc487126547)

[Referencias 4](#_Toc487126548)

# Introducción

Video es una captura, grabación, almacenamiento, y construcción de una secuencia de imágenes que representan escenas en movimiento. Digitalmente, el video se almacena mediante un estándar contenedor que almacena información de video, audio, subtítulos, capítulos, meta-datos e información de sincronización siguiendo un formato preestablecido en su especificación, y la información de video se comprime mediante la utilización de un códec, el cual permite su correcta reproducción.

**Códec**

Estos codecs son desarrollos en software, capaz de transformar un archivo con un flujo de datos o una señal. Los códecs pueden codificar el flujo o la señal y recuperarlo o descifrarlo del mismo modo para la reproducción o la manipulación en un formato más apropiado para estas operaciones.

**OpenCV**

Es una [biblioteca](https://es.wikipedia.org/wiki/Biblioteca_(programaci%C3%B3n)) libre de visión artificial originalmente desarrollada por [Intel](https://es.wikipedia.org/wiki/Intel_Corporation). Desde que apareció su primera versión alfa en el mes de enero de 1999, se ha utilizado en infinidad de aplicaciones. Desde sistemas de seguridad con detección de movimiento, hasta aplicaciones de control de procesos donde se requiere reconocimiento de objetos. Esto se debe a que su publicación se da bajo licencia [BSD](https://es.wikipedia.org/wiki/BSD), que permite que sea usada libremente para propósitos comerciales y de investigación con las condiciones en ella expresadas.[[1]](#Mediadrop)

**Reconocimiento de rostros con OpenCV**

Existen dos maneras de trabajar con OpenCV para el módulo de reconocimiento de objetos, caras, partes del cuerpo, etc. El primero es en tiempo real, si se necesita que se procese el video al tiempo que se guarda, se aplica el algoritmo de reconocimiento para cada frame capturado al momento, y en este caso, el resultado remarcarlo en un cuadro si se encontró, y posteriormente guardar el video. El segundo caso es cuando se requiere procesar un video ya almacenado, para este caso se lee el archivo de video y se aplica de igual manera el algoritmo frame por frame y se guarda en otro archivo el video resultante.

# Objetivo del código

El objetivo del módulo es presentar una aplicación que pueda capturar video desde una cámara instalada en un aula de clases o desde una cámara web y procesar los frames para aplicar algoritmos de reconocimiento de rostros, esto con la finalidad de analizar los patrones de comportamientos y gestos del profesor y mejorar su proxémica.

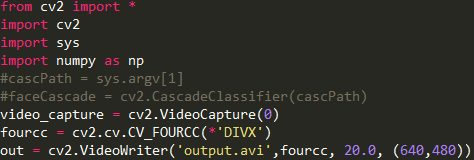
# Archivos

A continuación, se explicará de manera breve cada uno de los archivos con sus funciones con las que se realizó el código para reconocimiento de rostros por webcam.

## Face\_detect.py

En este archivo utilizamos la **libreria numpy** que agrega varias funciones que dan mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con estas variables. [[2]](#numpy)

Utilizando una de las funciones de OpenCV “VideoCapture(0)” y pasando como parámetro el Cero, se activa la webcam de la máquina para grabar.



*Imagen 1.1 VideoCapture.*

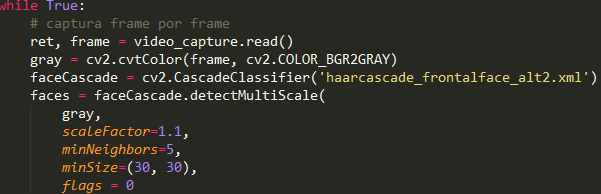
Posteriormente se utiliza el códec de video e la siguiente línea “cv2.cv.CV\_FOURCC (\*‘DIVX’).

Y se le asigna el nombre, la extensión y el ancho y largo del archivo de video que se generará.

De igual manera le pasamos la velocidad de frames y las dimensiones del video en el “VideoWriter”.

**CascadeClassifier**

Un CascadeClassifier básicamente le dice a OpenCV qué buscar en las imágenes. Existe variedad de cascadeClassifier en Internet y se puede encontrar el adecuado para determinadas especificaciones y utilizarlo. Pero la mayoría de ellos son para reconocer caras, ojos, oídos y boca. Se necesitó un clasificador en cascada que diga a OpenCV cómo reconocer una cara.

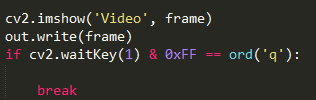


*Imagen 1.2 CascadeClassifier()*

En la siguiente imagen se muestra el código que ayuda a dibujar el marco alrededor el rostro reconocido:

**

*Imagen 1.3 marco de rostros*



*Imagen 1.4 cerrar video.*

La captura de video se terminará hasta que ingresemos la letra “q”, mientras seguirá desplegando lo que capture la cámara con la función “imshow” pasando como parámetros los frames capturados.

# Referencias

[1] <http://opencv.org/>

[2] <http://www.numpy.org/>

[3] <http://docs.opencv.org/2.4/doc/tutorials/objdetect/cascade_classifier/cascade_classifier.html>