

Wroclaw University of Science and Technology

Faculty of Electronics

10 de junio de 2020

Tomasz Baldach

Miguel Angel Vazquez Roa

Project Group: E10-04b

***Face Recognition and Detection using Tensorflow.***

Introduction to Computer Vision in Quality Control

***Face Recognition and Detection using Tensorflow.***

Objective

The objective is to implement a project capable to detect a face from a given image, then download images of famous people from google images and perform a face recognition with this data base.

Introduction to the detection and face recognition

If we think about making a detection in an image, we need to imagine that this is not at easy task, the first idea that we can have is try to apply a recognition algorithm to any possible sub-window in the whole image but this might result as a slow task and could have plenty of errors in the process. For this point is when we think about create detectors with a special purpose, these filters are supposed to quickly find likely regions in the images where some objects might occur.

One of the most know examples in this are is the face recognition which we can find in most of the digital cameras nowadays. The success of this detectors helps to the development of other aspects in the image processing sector.

What do we need to make the detection of a face and how does it work?

If we are planning to make a face detection in an image the first step that we need to follow is find the location and sizes of the faces in the whole image, in order to avoid at slow processing of the image by applying a face recognition algorithm to every pixel of the image we count with a huge variety of fast face detections algorithms developed in the past years in this case we use MTCNN (Multi-task Cascaded Convolutional Neural Networks) this is an algorithm implemented by tensorflow which consist of 3 stages. With this algorithm we can detect the bounding boxes of faces in an image along with their 5 point face landmarks.

In the first stage the input (The image given) is scaled down multiple times to build an image pyramid an each of these scaled versions passed through its CNN (Convolutional Neural Network). In the next stages we extract image patches for each bounding box and resize them. Then we forward them through the CNN of that stage. Stage 3 additionally computes 5 face landmarks points for each bounding box.

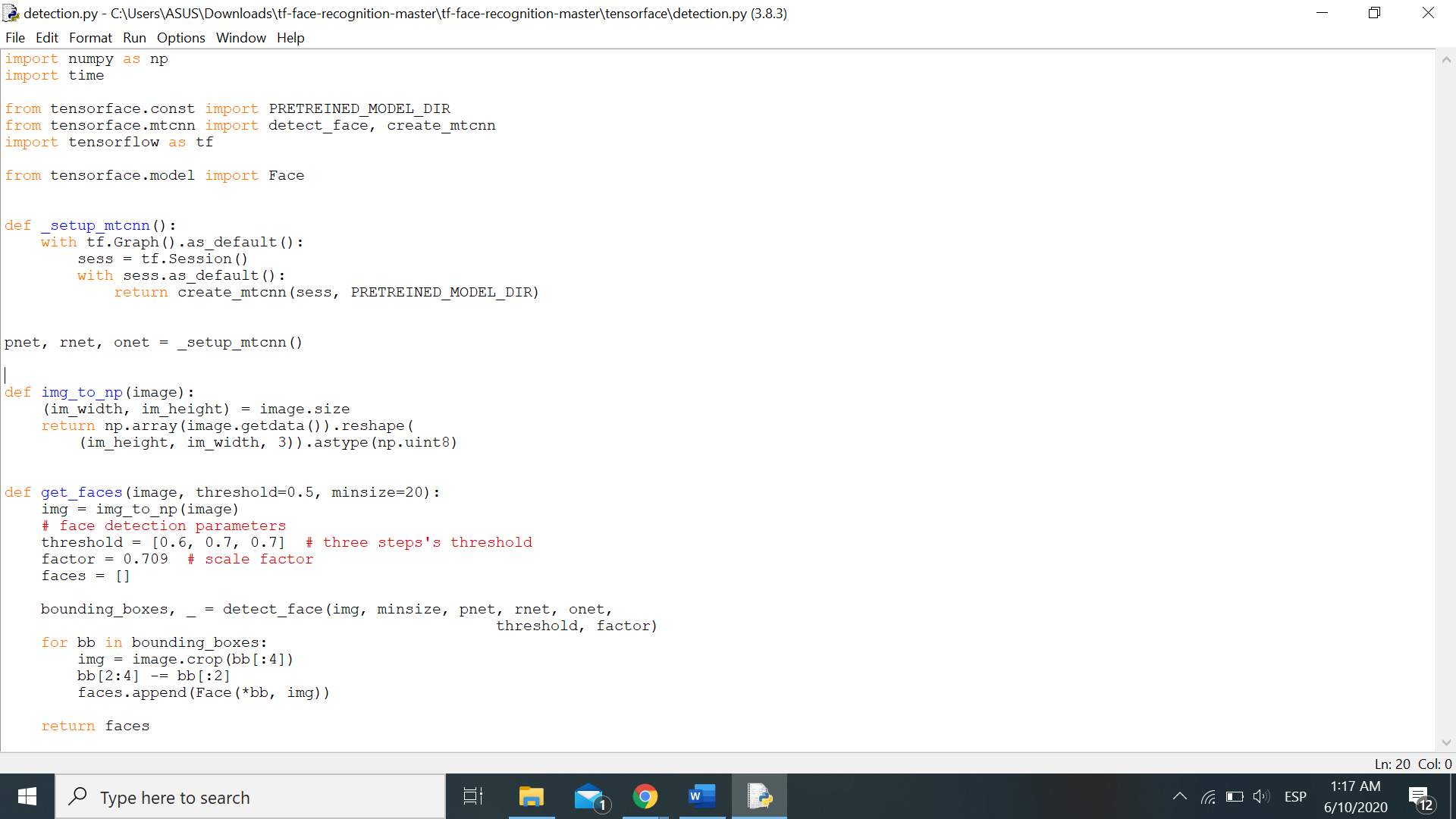


Figure 1 Code of the detection of the faces based on the tensorflow library.

In the last image we can notice the part of the code where the detection of the faces is made. In this process we obtain the Bounding boxes where the faces had been detected and we saved the bounding boxes for the next steps.

What do we need to make face recognition and how does it work?

Of the many recognition tasks that a computer can perform, face recognition might be the most successful one. The earliest approaches of face recognition involve finding the locations of distinctive image features such as the eyes, nose and mouth and then measure the distances between each one. Then the develop of the techniques let rely on comparing gray-level images projected onto lower dimensional subcases called eigenfaces. In this case as well that in the face detection FaceNet that directly learns a mapping from face images to a Euclidean space where the distances that we have there correspond to a measure of face similarity. Whit this space produce we can perform tasks as face recognition.

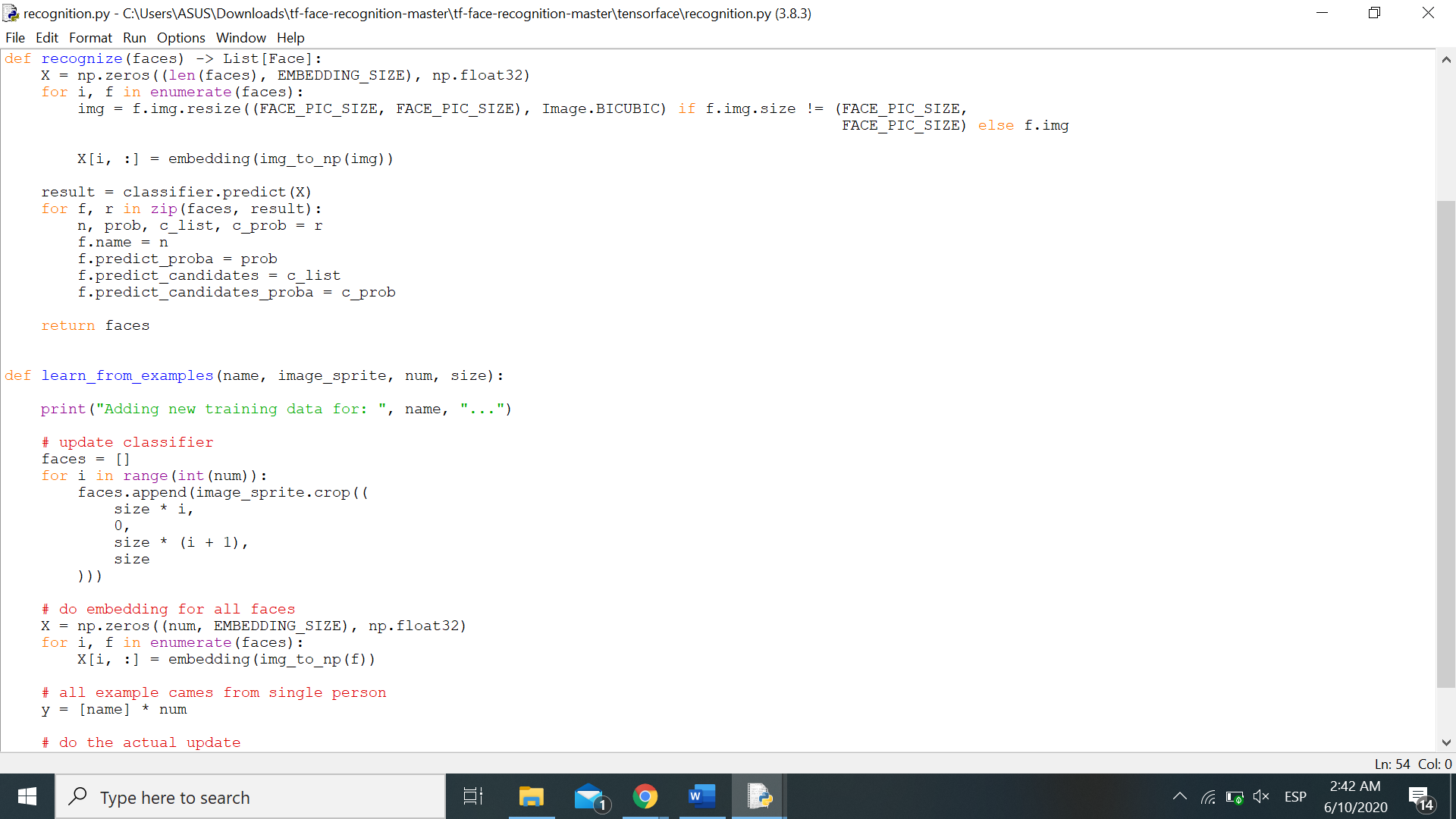


Figure 2 Figure 3 Code of the face recognition based on the tensorflow library.

In this part we generate pre-process the faces found in the previous part contained in the bounding boxes to create the mapping and extract the measures in the Euclidean space helped by the correspond function of the library. Then the program must compare the Euclidean spaces from the given image with the Euclidean spaces from the images previously learn.

The way that the images comparing function learn and obtain spaces to compare is through a search engine where it searches from famous faces in order to have a certain number of images to compare.

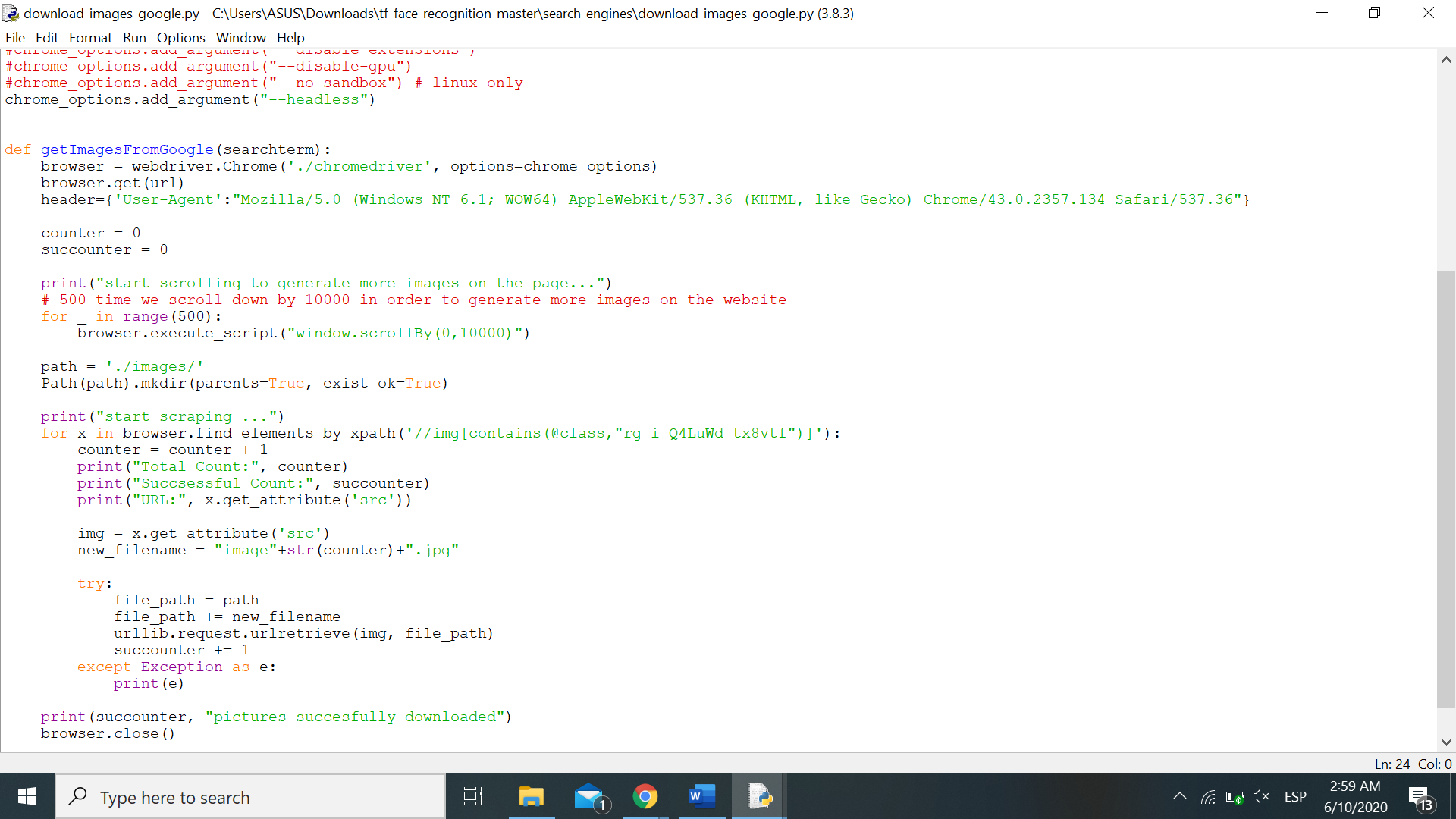


Figure 4 Code part of the search engine for extract images.

Result of the face recognition

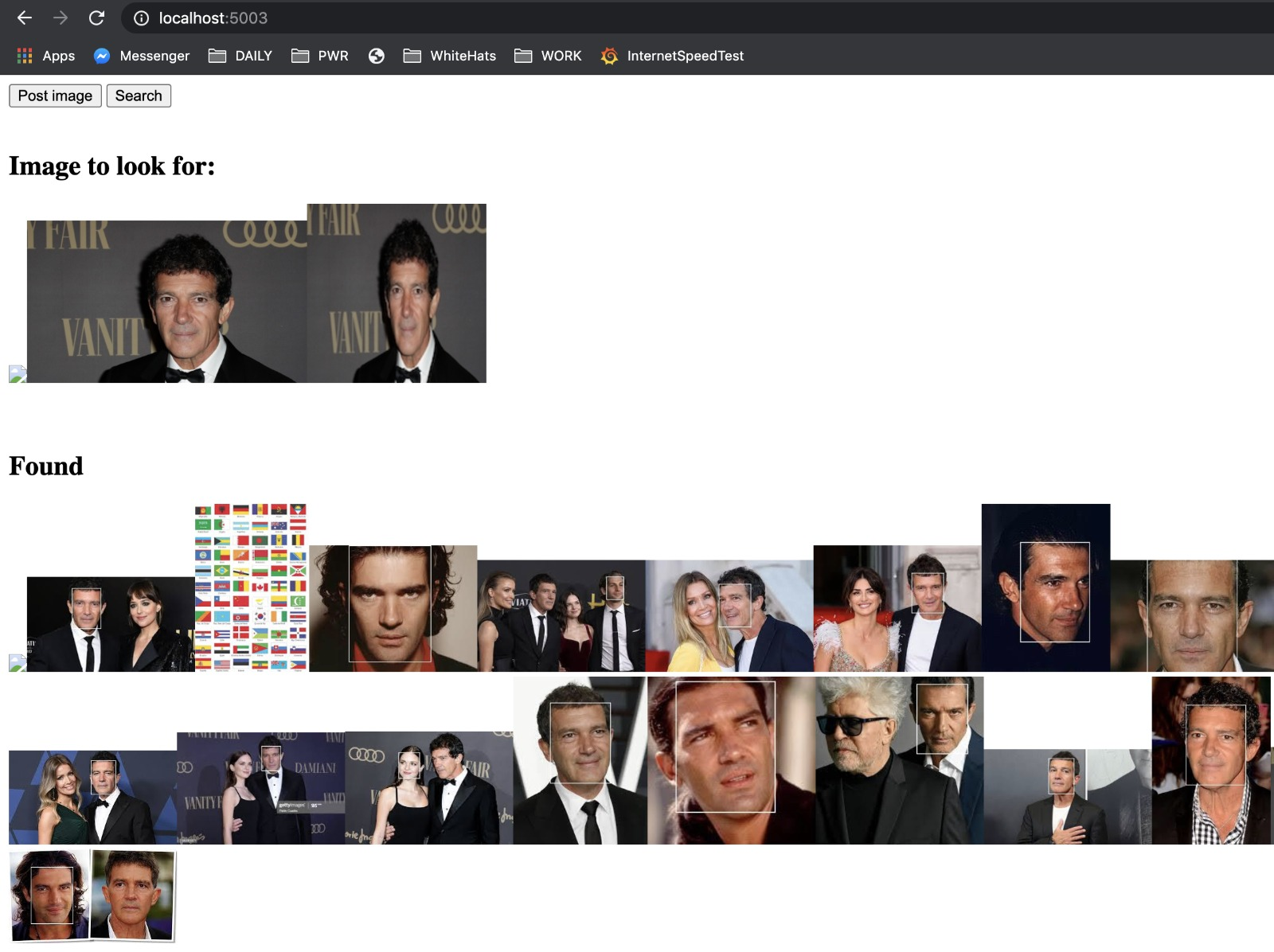


Figure 5 Result of the matches found by the project.

So as we can notice thanks of the images given as a source that we provide we can search in the images of the data to find matches in the rest of images and it give us many results as we can notice based in the measuring of the biomedical features of the face and comparing with the others bounding boxes in the Euclidean space.

Bibliography

[1] Computer Vision: Algorithms and Applications, 1st ed., 2010 Richard Szeliski, Microsoft Research [Last updated: Mar 26, 2020]. See Article History, <http://szeliski.org/Book/> [View on 01-Jun-2020]

[2] Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A

literature survey. ACM Computing Surveys, 35(4):399–358.

[3] FaceNet: A Unified Embedding for Face Recognition and Clustering, [Last updated: June 17, 2015]. See Article History.- https://arxiv.org/abs/1503.03832 [View on 01-Jun-2020]

[4] Face Recognition Using Knn & OpenCV, [Last updated: October 26, 2019]. See Article History.- https://medium.com/analytics-vidhya/face-recognition-using-knn-open-cv-9376e7517c9f [View on 01-Jun-2020]