**Milestone Report # 1**

**Group H**

* Saurabh Laltaprasad Gangwar
* Brayan Leonardo Gil Guevara
* Rohit Kumar
* Marzieh Mohammadi Kokaneh
* Eduardo Roberto Williams Cascante

# 1. Data collection

## 1.1 Data source

Since the goal of our project is to build a model which tries to recognize our faces, we took one hundred pictures (each group’s member) from our cellphone’s personal gallery. This set of pictures was made up as follows: ninety pictures with our faces in first plane and ten pictures without our faces. Those ten pictures were intended not to be labeled.

1.2 Data preparation

For the data preparation process we performed two main steps: cropping and resizing.

The purpose of cropping was to set pictures in an exactly square shape, it means, with the same height and width number of pixels. This was necessary for the resizing step. In addition, we first cropped them to make faces take up a third of the photo at least, in other words, we cropped elements in the pictures other than our faces.

So as to reduce pictures weight and not to collapse the following processes (annotation, augmentation, etc.) we had to resize all the pictures to 480 x 480 pixels. Additionally, augmentation process needs all photos with the same shape because it performs a little random cropping and also receive the normalized labels coordinates as input. These coordinates are strictly related to pictures width and height.

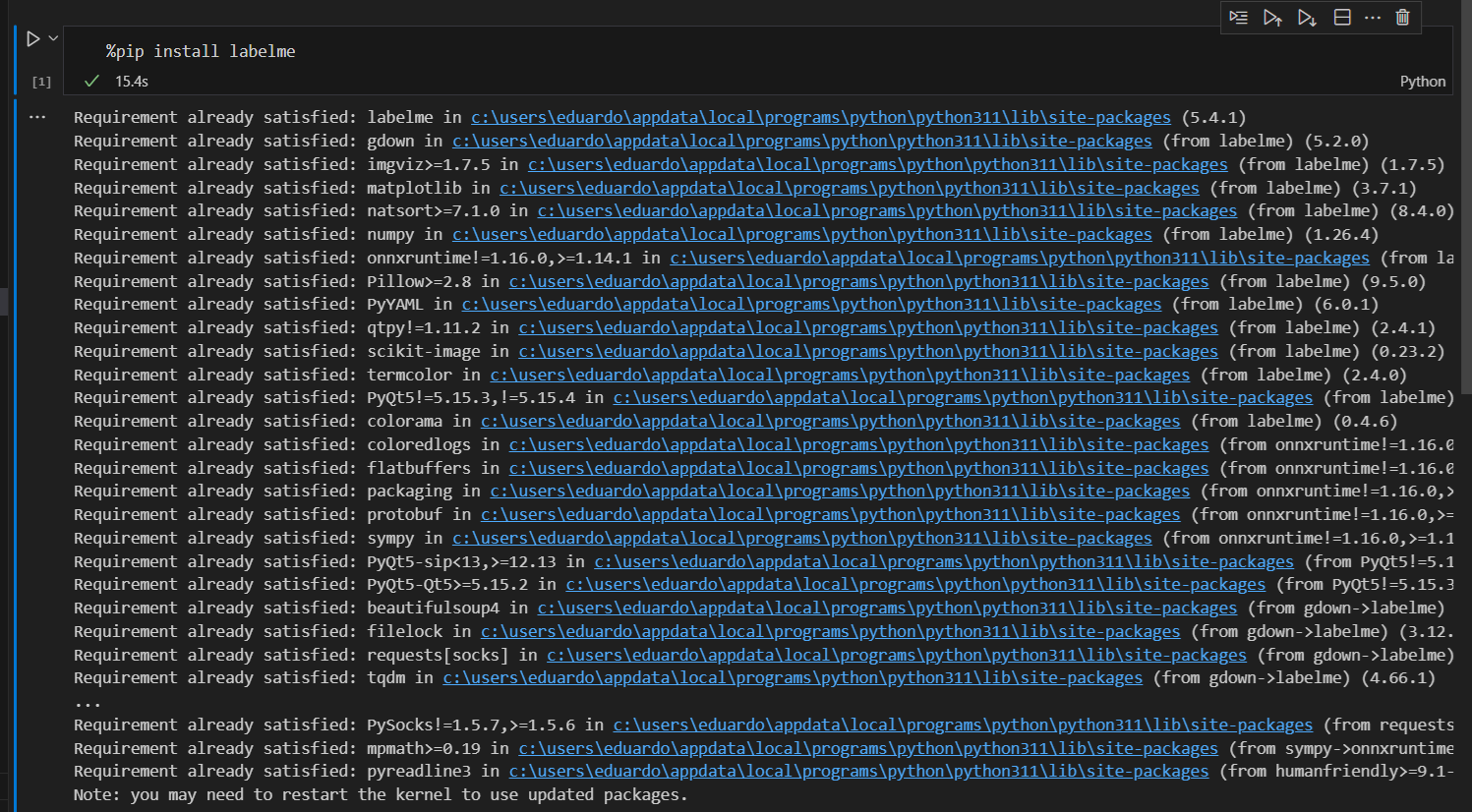
Both cropping and resizing steps were made manually by using Paint and Adobe Photoshop.

# 2. Data annotation

2.1 Labelme

We performed data annotations by using a tool called Labelme. We set up and ran Labelme from a python notebook, with the following commands:

* **%pip install labelme**
* **!labelme**

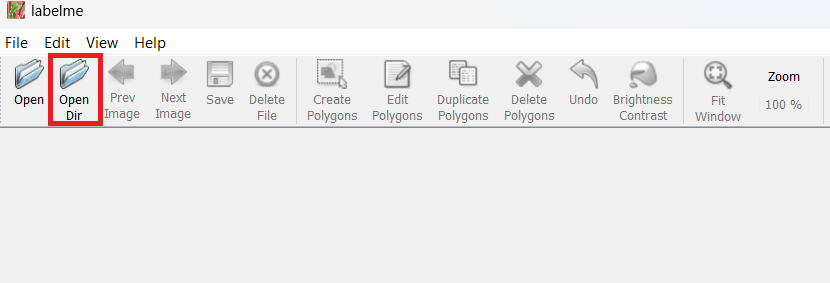


**Figure 1** Labelme installation

2.2 Data labeling

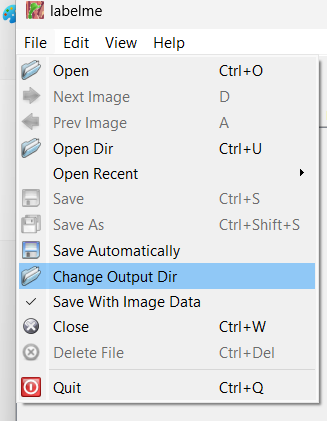
In order to label correctly each face in pictures we defined some settings in Labelme:

* **Open Dir:** It refers to the folder which contains all cropped and resized photos.



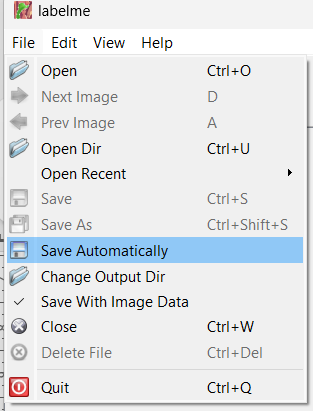
**Figure 2** Labelme – Open Dir

* **Output Dir:** It refers to the folder where Labelme will save labeling information files. They are files in json format which contain information related to the coordinates of labeling as well as classes names, etc.



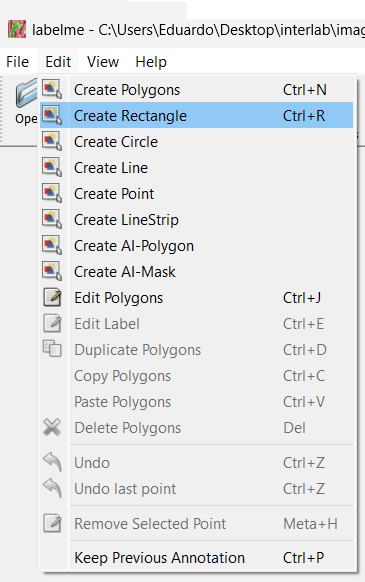
**Figure 3** Labelme – Open Dir

* **Saving mode:** Additionally, we setup Labelme for it to automatically save each label we did, instead of pressing Save button for each file.



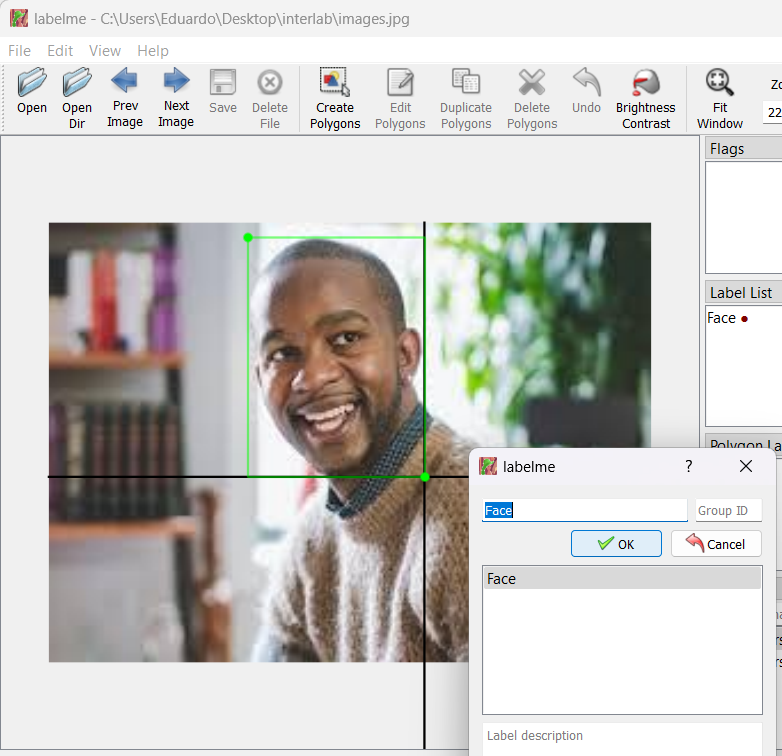
**Figure 4** Labelme – Saving mode

* **Shape:** Finally, we selected the shape (rectangle) we wanted to draw for labeling.



**Figure 5** Labelme – Shape

Once Labelme was set up, the next step was to label faces on each photo, as shown in the next figure:



**Figure 6** Labelme – Labeling

For each labeling we did Labelme saved the related data in a json file with the same name as the picture file. This file contains some information such as version, picture filename, image weight, image height, however the most important information are the label and the points, this latter contains the coordinates of labeling.



**Figure 7** Labelme – json file