

Face Mask detection

NAME -> Gaurav trivedi

ROLLNO ->2013685

SECTION ->ML (Machine Learning)

COURSE ->BTech

BRANCH ->Cse ML&AI

SEMESTER ->5th

PROBLEM STATEMENT

FACE MASK DETECTION

In the present scenario due to Covid-19, there is no efficient face mask detection applications which are now in high demand for transportation means, densely populated areas, residential districts, large-scale manufacturers and other enterprises to ensure safety. Also, the absence of large datasets of **‘with\_mask’** images has made this task more cumbersome and challenging.

DATASET

This dataset consists of **3835 images** belonging to two classes:

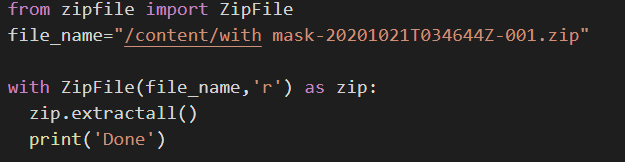
* **with\_mask: 690 images**
* **without\_mask: 686 images**

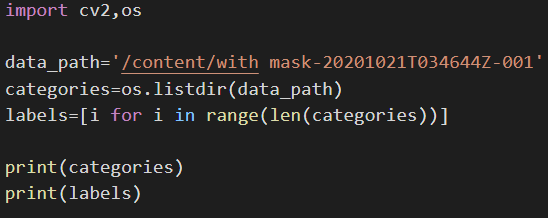
The images used were real images of faces wearing masks. The images were collected from the following sources:

* **Kaggle datasets**
* **RMFD dataset**

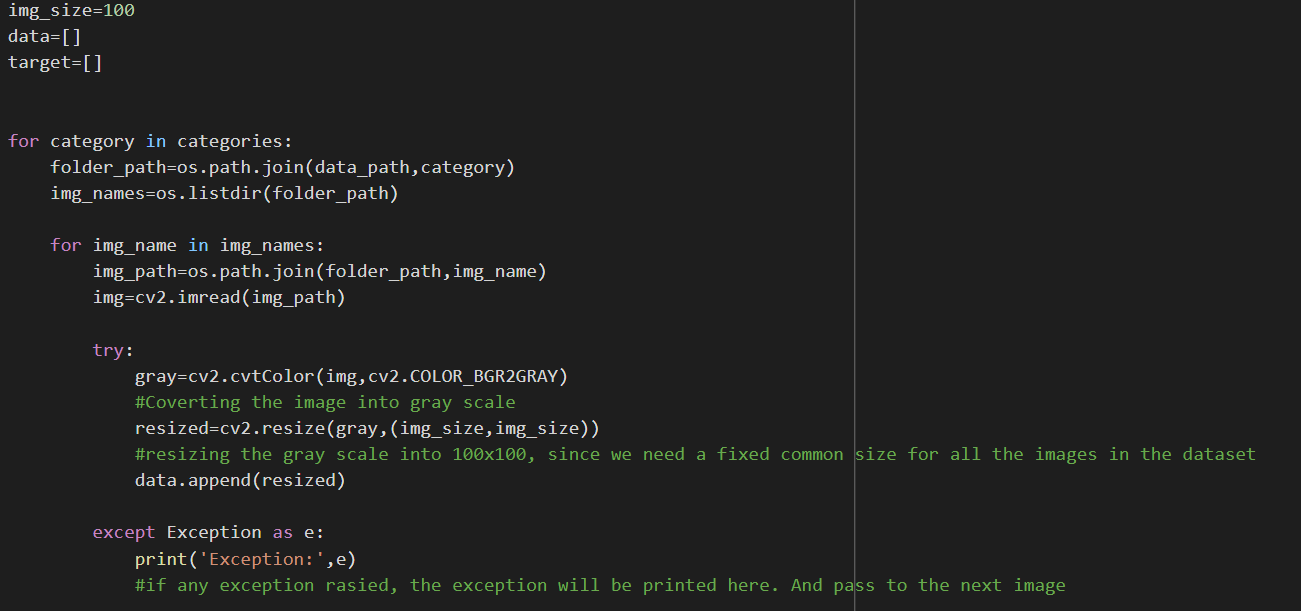
APPROCH

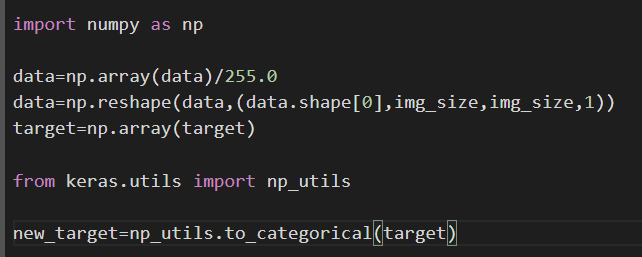
In general, Image-based methods rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face and non-face images. The learned characteristics are in the form of distribution models or discriminant functions that is consequently used for face detection. In this method, we use different algorithms such as Neural-networks, HMM, SVM, AdaBoost learning. In the coming section, we will see how we can detect faces with MTCNN or Multi-Task Cascaded Convolutional Neural Network, which is an Image-based approach of face detection

**EXPLANATION OF THE CODE**

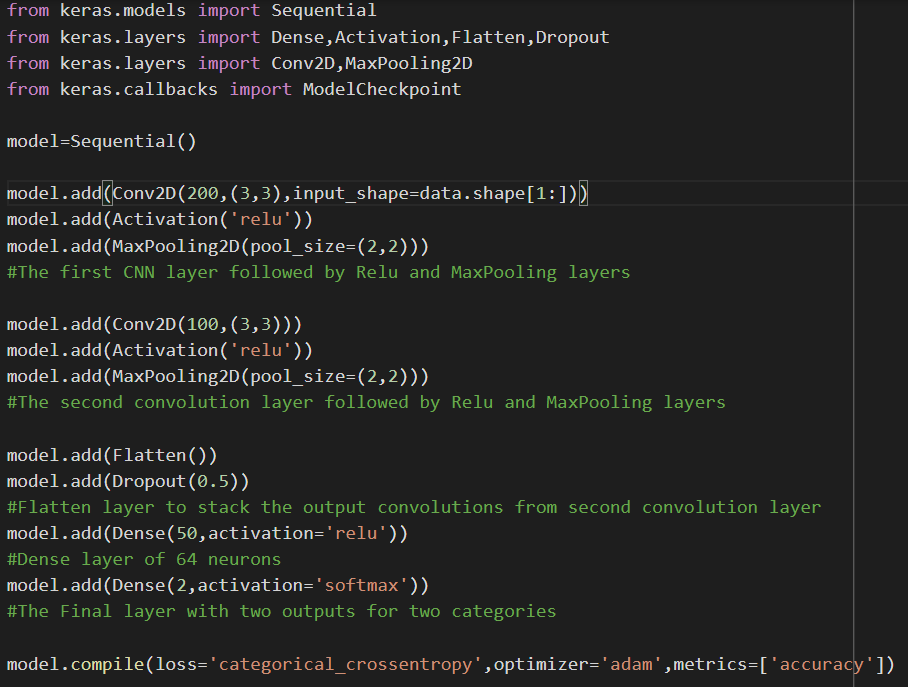
In this snippet we have extracted the file form the zip which contains our data set as uploading a normal folder was a tedious task uploading 690 pictures form the local machine.

In this snippet we have given the path for the data set for the preprocessing of the data set which we will be doing later in the code. In this cell we have printed the categories and the labels present in the dataset.

In this snippet in the very first for we will be creating a folder like for the images in which we have people wearing the mask in the second for loop we are going through all the images in that folder and we will be reading those images with the help of cv2in the try block first we will convert the image in the gray scale and the convert the gray scale into 100x100, as we need a fixed common size for all the images in the dataset. Then we have resized the images and stored them into the data list.

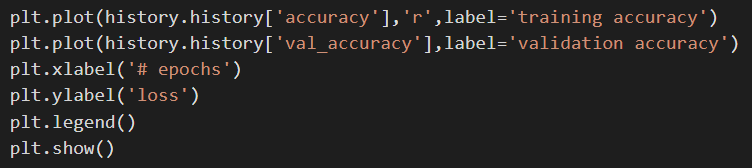


In this snippet we have normalized the image by dividing them by 255 which will convert the pixel range in 0 and 1. Then we are going to convert it into 4 dimensional array. Then we have converted the target into numpy array. Then we have converted the target into the categorical data as we are going to need them. Lastly in the preprocessing we are going to save the data and the target.

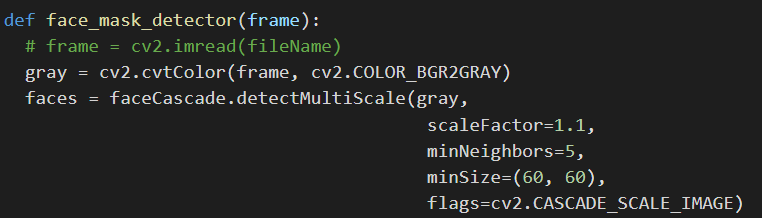


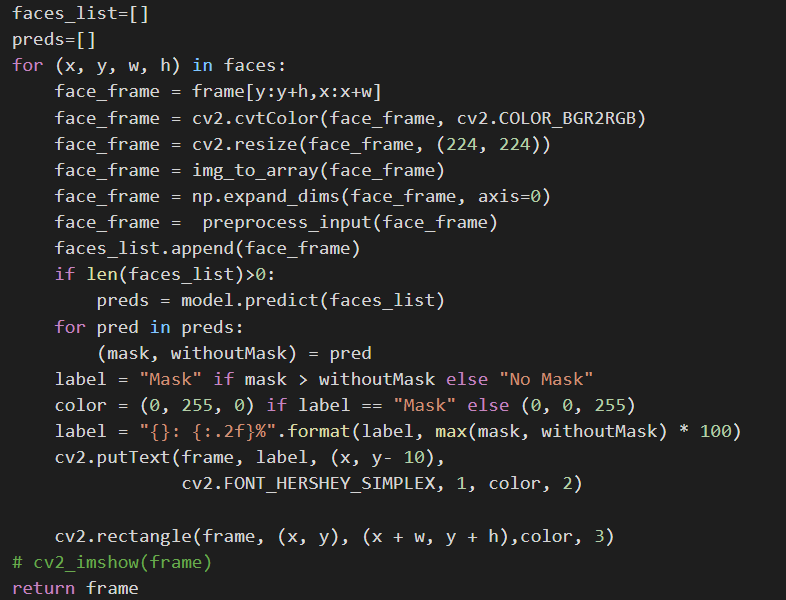
In this snippet we have first imported some of the libraries of the keras which we will be using in our code later on. In the first convolutional layer we have used 200 kernels of size 3x3 followed by the reluand MaxPooling layer. Then in the second layer we have used 100 kernels of size 3x3 followed by the relu and maxpooling layer. Then we have flatten layer to stack the output convolutions form second convolution layer. After doing this we have dense the layer of 64 neurons. After which we have the final layer with two output for two categories.

There we have have created a check point so that we can get the model with the highest accuracy. The model with the highest accracy will be selected for the further process in the project as it will give us the highest accuracy in that video.

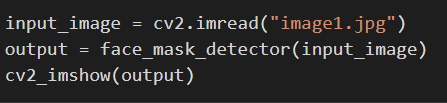


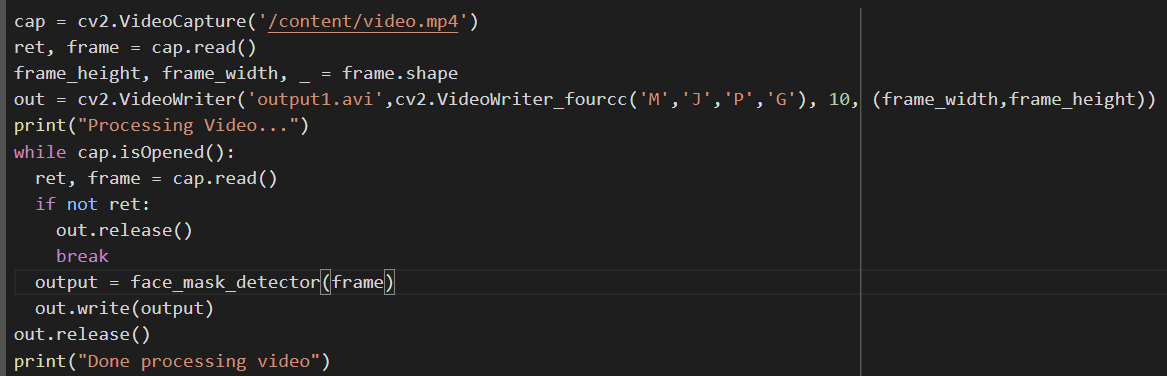
There we have developed a plot which is showing us the training accuracy and the validation accuracy. With this we are done with our CNN training now the only thing left for us is to detect the mask.

Now we have started with our third code in which first we have imported the model which was the most accurate one. For that I had trained my CNN model 3-4 times and I got the result as that **mask\_recog.h5** has the most accuracy

Here the first thing we have done is convert the image into gray scale which we have done earlier also but it is ok to do it again as it will not cause any harm on the data. Then we have used faceCascade which is harcascade to detect the faces in the images. Here we have detectMultiScale so that we can detect multiple faces in a single frame.

After we have loaded the images we send them to this particular cell in which we have resized our images just to be sure that we do not have any sort of error in the images while our model is detecting for the mask.



Now the moment of truth is here if our model is working or not we have called out our mask detection program and if everything goes right we will be getting our result as in the image the lady is wearing the mask we will do the same for the not wearing a mask.

When we are done with that we will see if our model is success full in video as well.

**THANK YOU**