**Face Mask Detection**

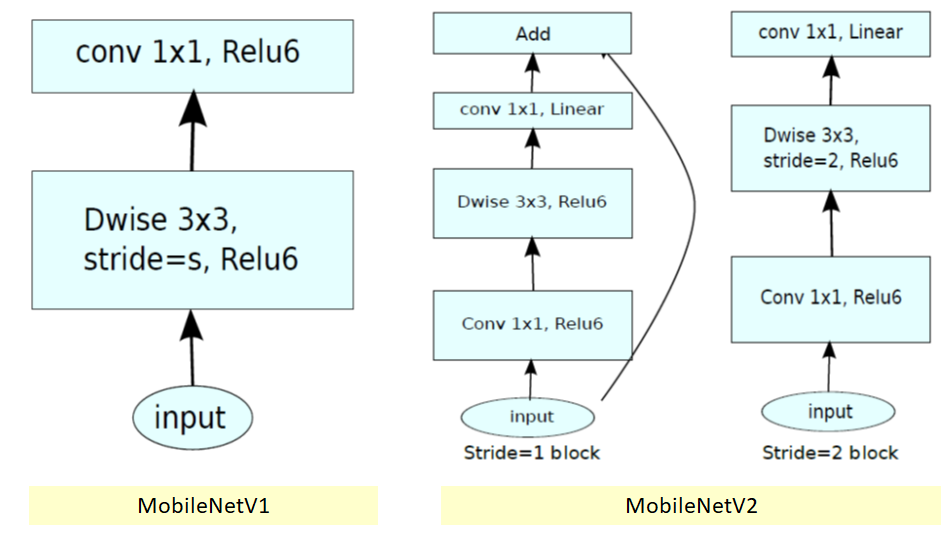
Face Mask Detection is a project based on Artificial Intelligence . In this we detect people with or without mask . In making of this project we have two phases .

* Train model using Convolution or any pretrained model which detect face masks in images .
* Then, Detect faces in video or images and get prediction from our trained model

In this Project I used Transfer learning to train model on dataset in which I used pretrained model MobileNetv2 , Now question is what MobileNetV2 is ? what is the Strucutre of it ?

**MobileNetV2**

MobileNets are small, low-latency, low-power models parameterized to meet the resource constraints of a variety of use cases. They can be built upon for classification, detection, embeddings and segmentation similar to how other popular large scale models, MobileNetv2 is a Significant Improvement over MobileNetV1 . The Structure of MobileNetV1 and V2 is below .



Now , Come to Code Explanation

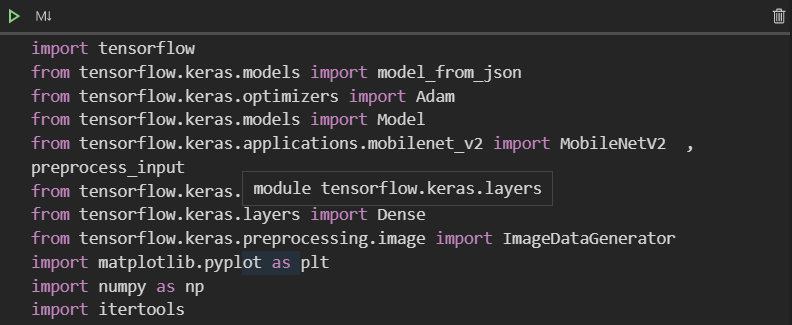
**Dataset**

In the Dataset we have two types of images one who wore mask and other which do not have mask.

Training Dataset consist of 4296 images in which 2455 images have label with mask and 1841 images are of without mask faces. and Validation Dataset consist of 300 .

**Import :**

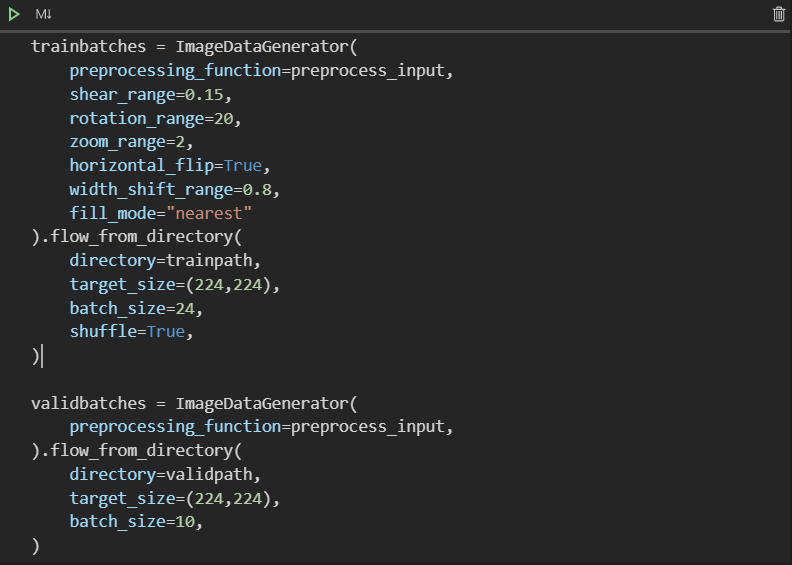
First I import Required modules like tensorflow , keras ,optimizer , layers and pretrained model(MobileNetV2)



**Loading Data And Augment Image** :

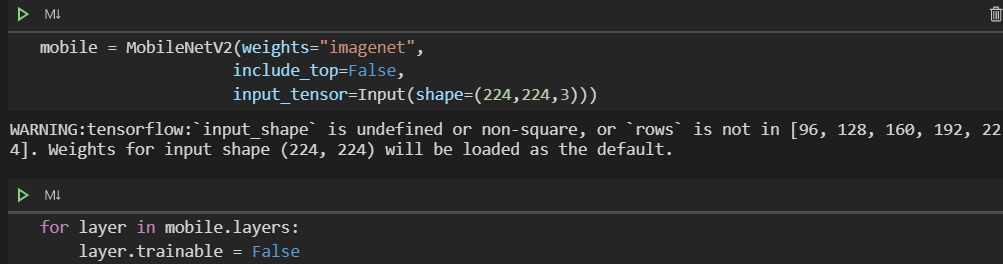
Now I load all images for training and preprocess using MobileNetv2 preprocess\_input through which the images are ready as mobilenetv2 require . we also Augment our images through which we can get more data and variety in data. we add augmentation technique like zoom , horizontal and vertical flip , rotation .

After this step we have training and validation batches which mobilenetv2 require as input .



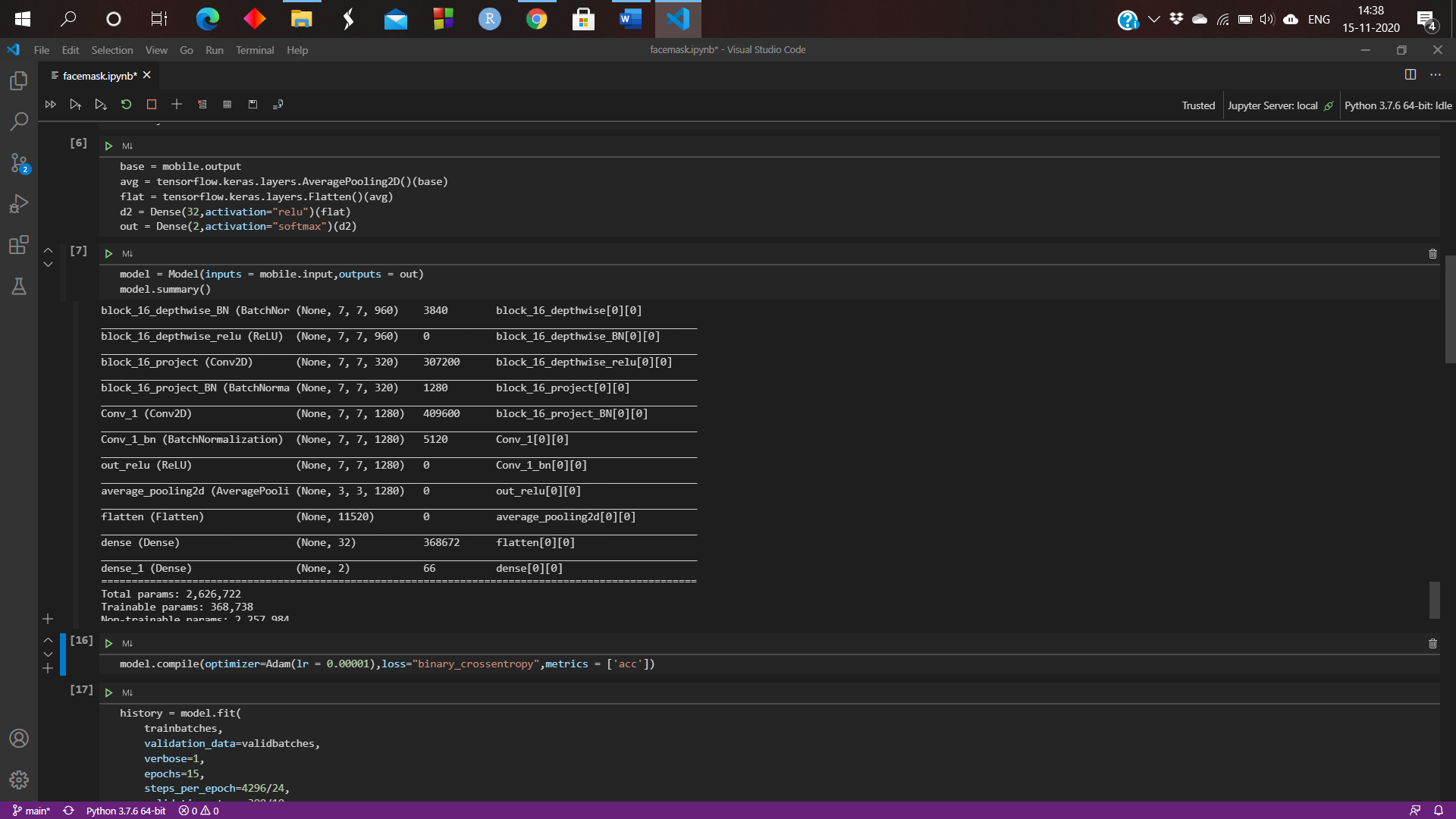
**Loading Pretarined Model**  :

As I previously write I Used MobileNetV2 . So , I download weights of model and create a object of MobileNetV2 type . after we freeze the layers of our pretrained model . through which layer weights can’t be modified when our model is in training .



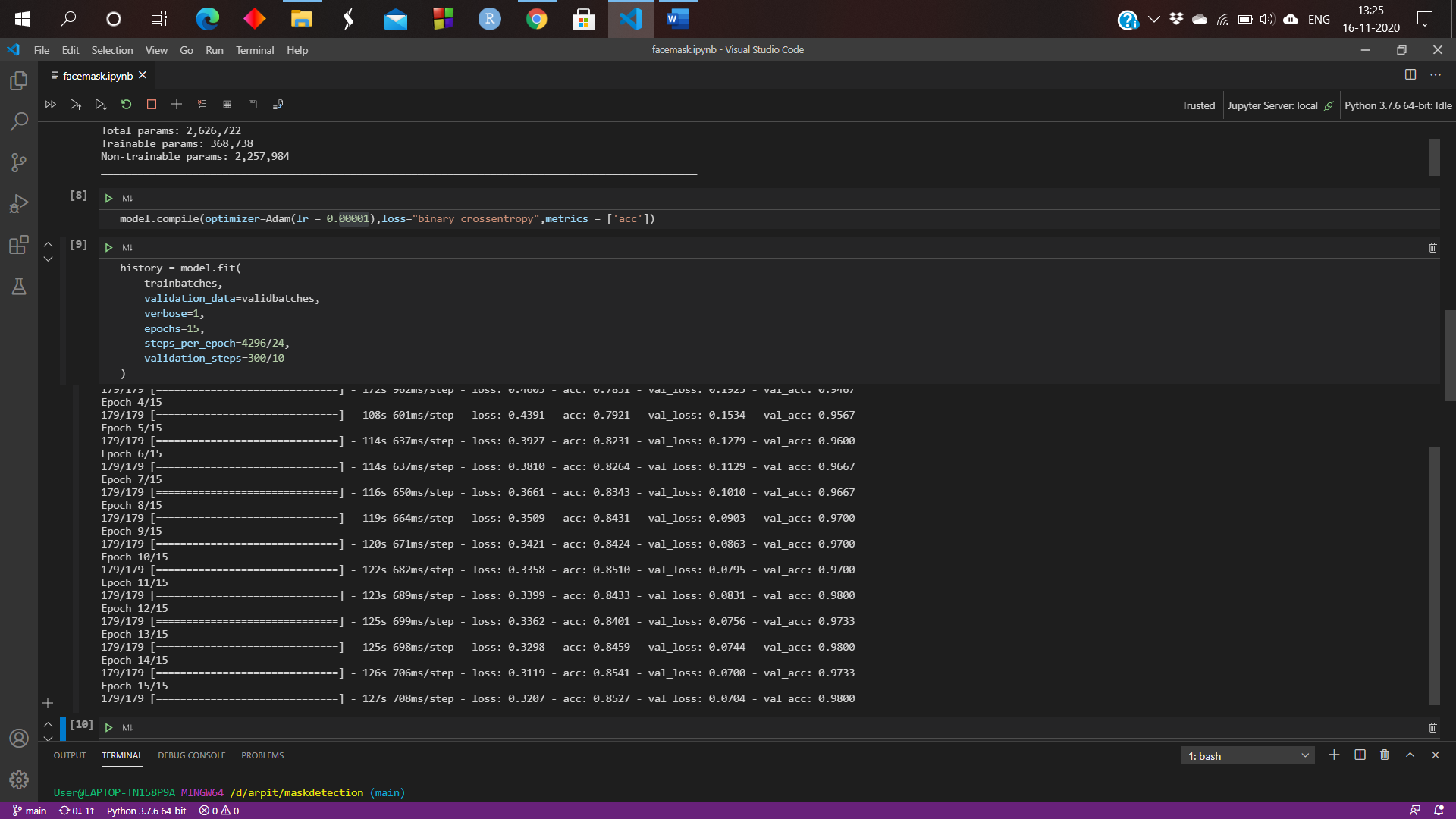
**Adding some layers Model**  :

Adding Some Layer at the end of the model to achieve good accuracy or save model from overfitting . At last add one fully connected which contains neurons equal to number of labels we have (In this case we have 2 labels mask or no mask).



**Compile and Train**  :

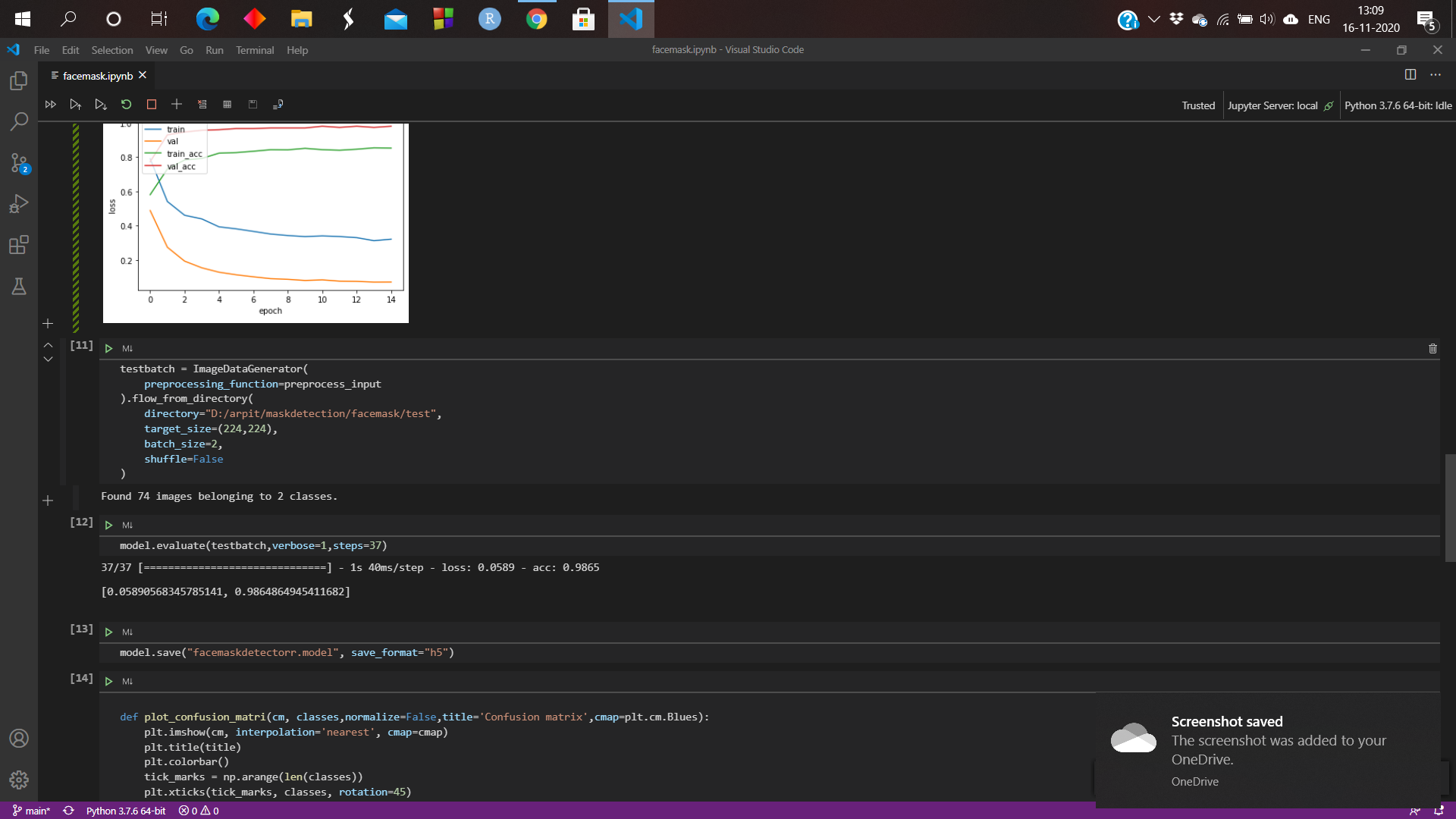
Now it’s time to train the model but before training we have to define loss function and optimizer i.e., compiling a model . I used Adam Optimizer with learning Rate 0.00001 and loss function binary\_crossentropy . then after I trained the model with 15 epochs and validate it on validation data (which contains 300 images) .



**Validation Phase**  :

Now we have to validate model on test data to check wheather model will work good on real time data or not . so , we have 74 images of 2 classes as test dataset after evaluating it on test dataset I got good results .

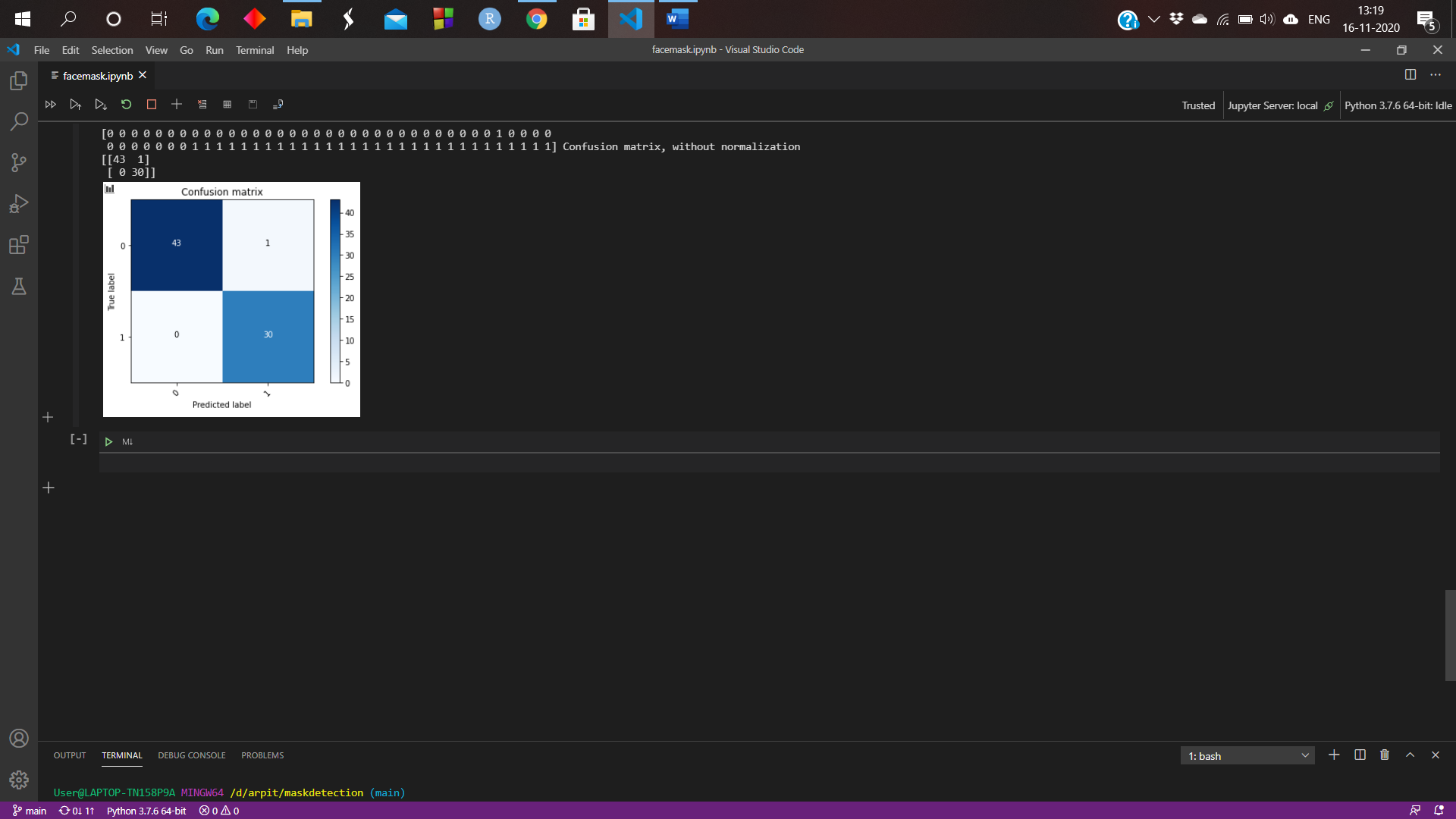
Loss : 0.058



**Confusion Matrix :**

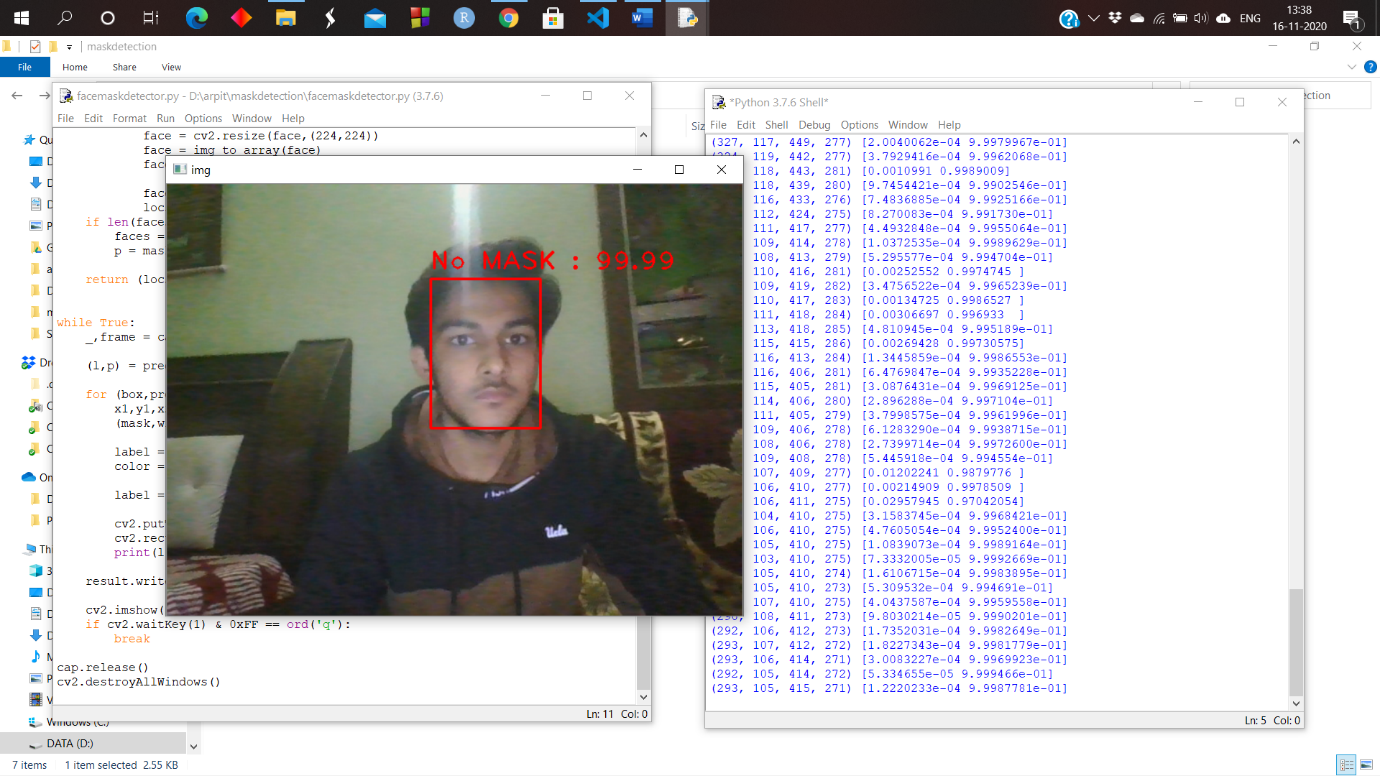
A confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known.

After Plotting Confusion Matrix what I see Model only predicts 1 image wrong else all images are correctly predicted . so model will work with realtime .



As I got good results so it’s time to implement it with camera to work in real time .

Thereafter , Using Opencv I implemented this model with my webcam . using caffe model I predict faces in image then after send it to our trained model to predict wheather person wering mask or not .



**Conclusion**  :

Our current method of detecting whether a person is wearing a mask or not is a two-step process:

1. Perform face detection
2. Apply our face mask detector to each face