THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

DATS 6203_10: Machine Learning 2
Amir Jafari

Machine Learning 2 Final Project

Face Mask Detection

Group 1

- Osemekhian Ehilen
- Atharva Haldankar

Introduction

With the eruption of Coronavirus disease 2019 (COVID-19) in Wuhan, China, in December 2019; the world has not remained the same. The importance of face masks became mandatory to dampen the spread through the human respiratory channels.

This project will show and compare machine learning models on facemask prediction.

Description Of The Dataset

- The dataset is from kaggle, titled Face Mask Detection with 853 images belonging to 3 classes as well as their bounding boxes with the face mask location.
- The classes of images are:
  1. With mask
  2. Without mask
  3. Mask worn incorrectly

Scope of Work

1. Loading dataset
2. Pre-processing dataset
3. Training with pretrained models
4. Training without pre-trained models
5. Visualization of the model history
6. Prediction using the model saved

Data Augmentation

- For both the models we have done the following augmentation:
    1. Rescaling
    2. Horizontal flips
    3. Zoom range
    4. Shear range
    5. Width shift range
    6. Height shift range
    7. Rotation range

Training Algorithm | Framework

The **Adam** optimizer was used as our training algorithm with **TensorFlow** as our framework.

Training with Pre-trained model

Xception, **VGG16** and **ResNet50** pre-trained models were leveraged in predicting the three classes of our dataset. The models had various input shapes:

1. Xception: 299 by 299
2. ResNet50: 224 by 224
3. VGG16: 224 by 224

Training with Pre-trained model

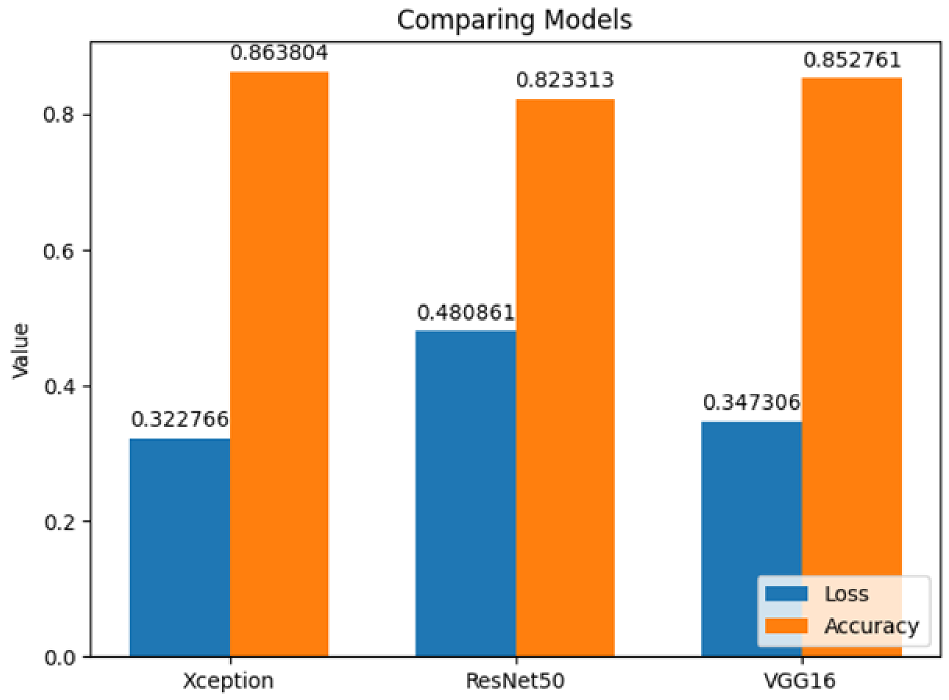
The loss was calculated by cross entropy and the measure of performance was accuracy score.

A batch size of 16 for its mini-batches was used.

Also, the Performance Scheduler was used in fine tuning the learning rate during training.

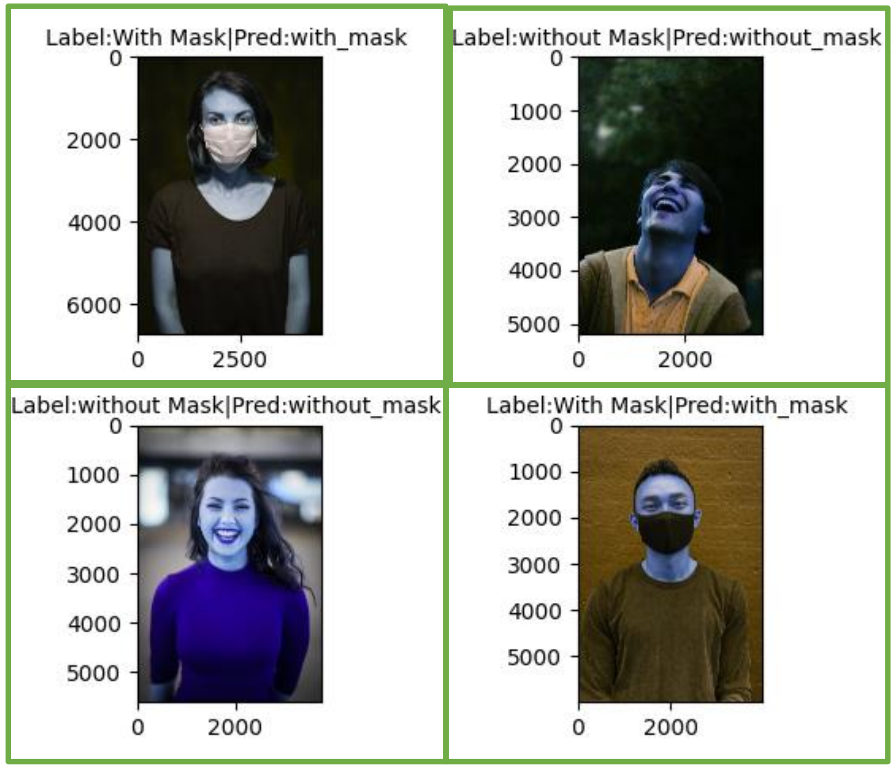
Results with Pre-trained Model

- We have made splits on the dataset into training, validating and test set.
- After training using Xception, ResNet50 and VGG16 pre-trained models, Xception came out top.



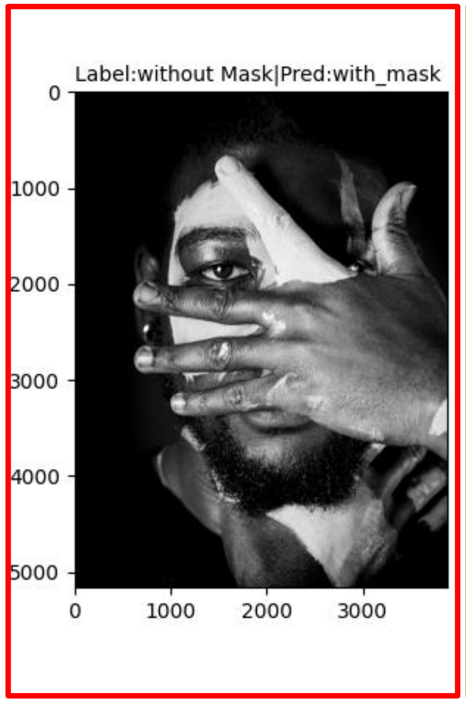
THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

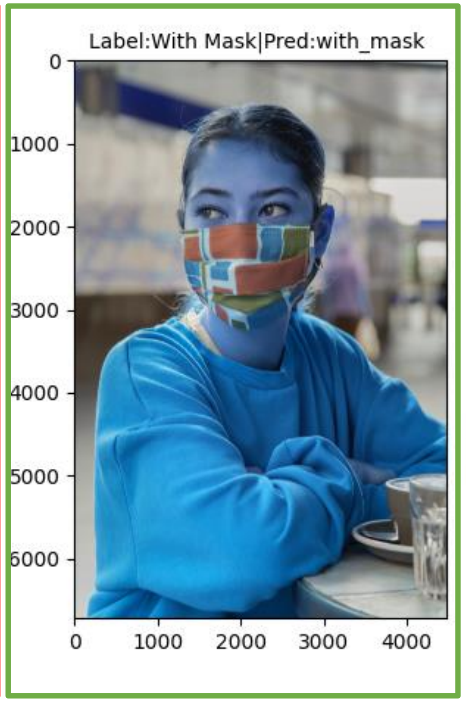
Results with Pre-trained Model


Label:With Mask|Pred:with_mask
Label:without Mask|Pred:without_mask
Label:without Mask|Pred:without_mask
Label:With Mask|Pred:with_mask

- With Xception having the best score we tested some images from unsplash.com to see its performance.
- This performs pretty accurately.

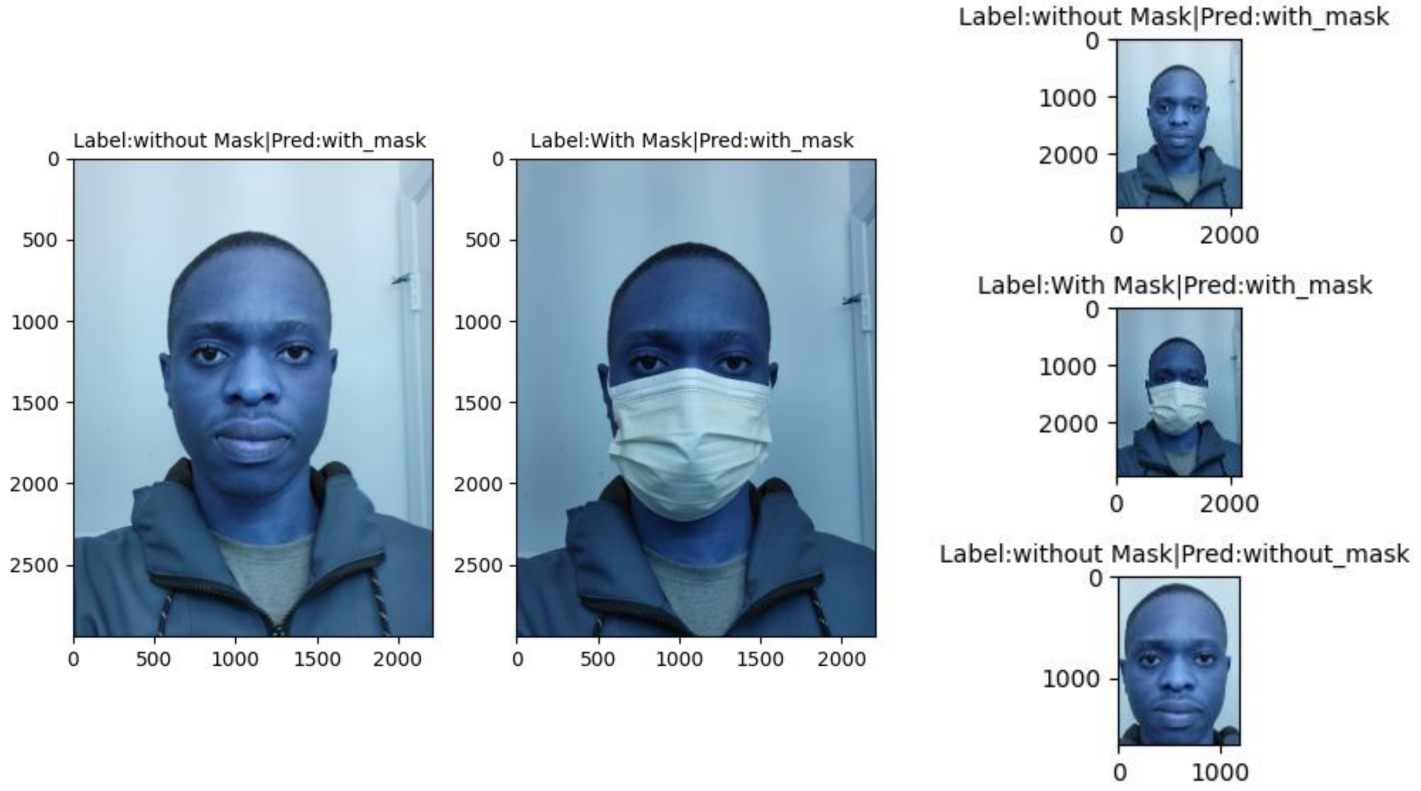
Results with Pre-trained Model





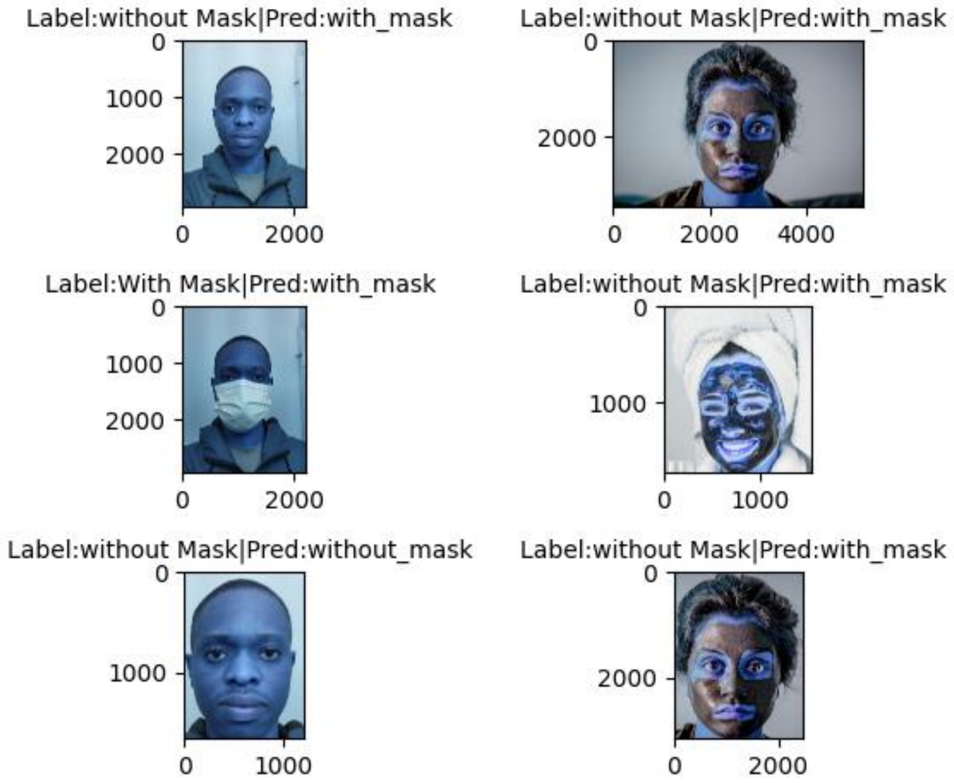
- We also gave a tricky image to the model and it performed strangely.
- The model seems to think that the person's hand is the facemask on the left image.

Results with Pre-trained Model


Label:without Mask|Pred:with_mask
Label:With Mask|Pred:with_mask
Label:without Mask|Pred:with_mask
Label:With Mask|Pred:with_mask
Label:without Mask|Pred:without_mask

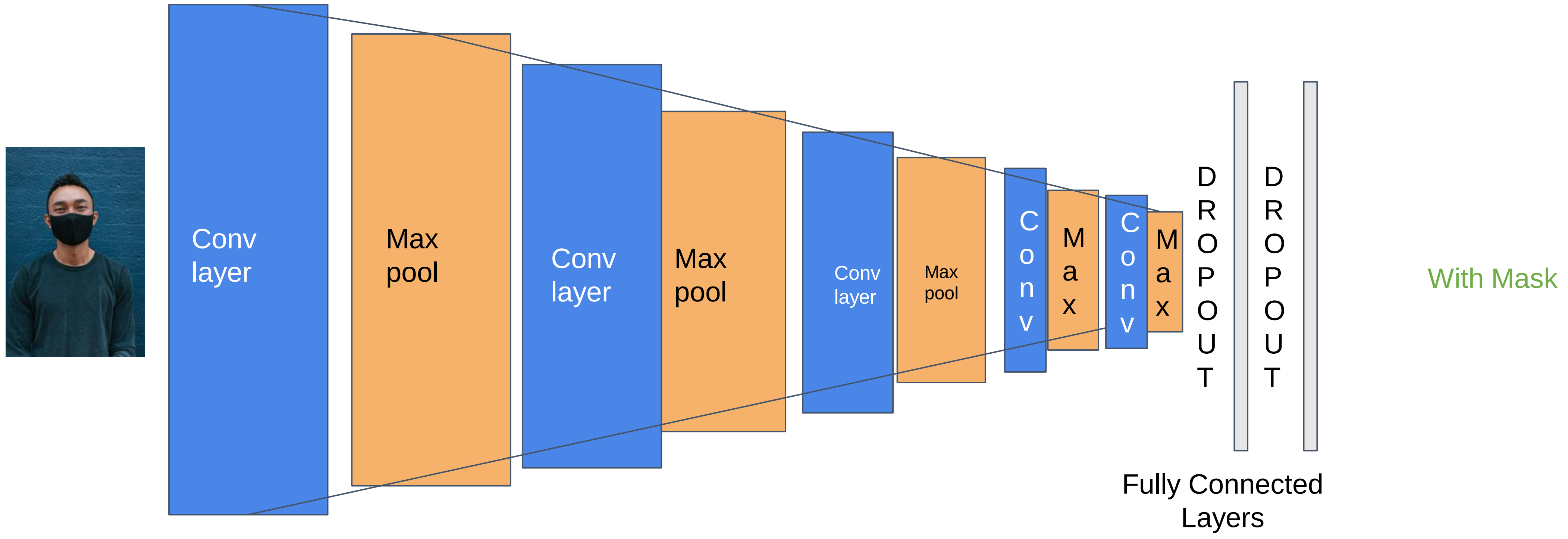
The model behaves well when the person's face is detected/cropped out.

Results with Pre-trained Model


Label:without Mask|Pred:with_mask
Label:without Mask|Pred:with_mask
Label:With Mask|Pred:with_mask
Label:without Mask|Pred:with_mask
Label:without Mask|Pred:without_mask
Label:without Mask|Pred:with_mask

The model cannot differentiate between a facemask and beauty mask.

Architecture for Custom CNN Model



Train without Pre-trained model

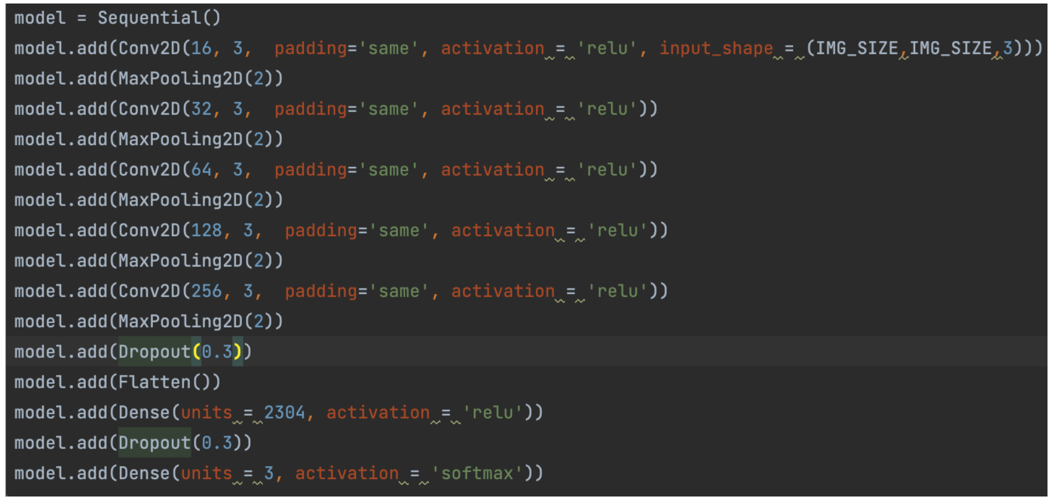
- We used Adam as the optimizer for this model, as we achieved the best results with this Optimizer.
- Categorical Cross entropy was used as the loss function.
- Performance Scheduler for fine tuning the learning rate.
- The learning rate stays constant at 0.001 throughout with image size = 100, batch size = 8 and epochs = 50.

Train without Pre-trained model

- We have also considered all the three classes for this model
- The annotations of these images have given the dimensions to extract the faces from the pictures.
- We extracted the faces using the data separated all the images of different categories into different folders to train the model.

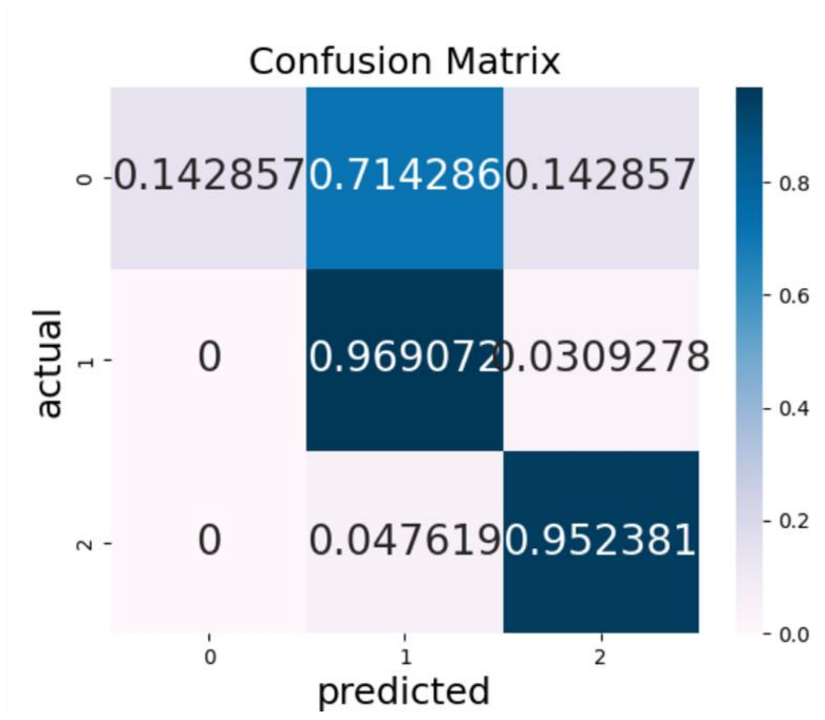
Train without Pre-trained model

- Then we created a sequential model using 15 layers which are shown in the picture below.



```
model = Sequential()
model.add(Conv2D(16, 3,  padding='same', activation = 'relu', input_shape = (IMG_SIZE,IMG_SIZE,3)))
model.add(MaxPooling2D(2))
model.add(Conv2D(32, 3,  padding='same', activation = 'relu'))
model.add(MaxPooling2D(2))
model.add(Conv2D(64, 3,  padding='same', activation = 'relu'))
model.add(MaxPooling2D(2))
model.add(Conv2D(128, 3,  padding='same', activation = 'relu'))
model.add(MaxPooling2D(2))
model.add(Conv2D(256, 3,  padding='same', activation = 'relu'))
model.add(MaxPooling2D(2))
model.add(Dropout(0.3))
model.add(Flatten())
model.add(Dense(units = 2304, activation = 'relu'))
model.add(Dropout(0.3))
model.add(Dense(units = 3, activation = 'softmax'))
```

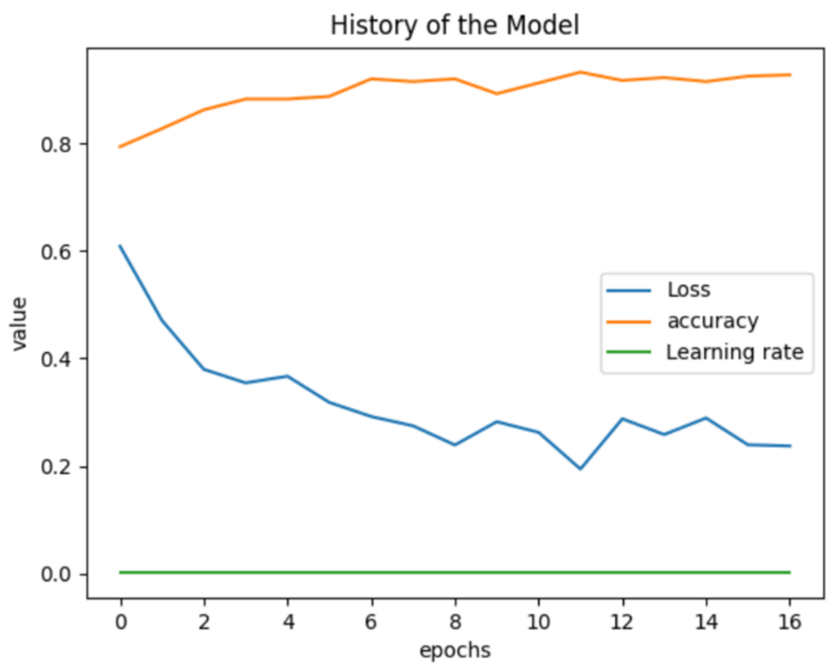
Results without Pre-trained Model

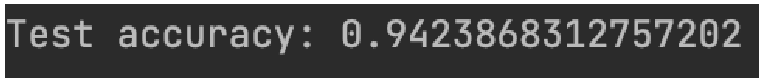


- From the confusion matrix we can see that
  1. label 1 : With_mask have the highest prediction accuracy
  2. Label 2: without_mask has second highest
  3. label 0: mask_worn_incorrectly has the least accuracy.

THE GEORGE WASHINGTON UNIVERSITY
WASHINGTON, DC

Results without Pre-trained Model

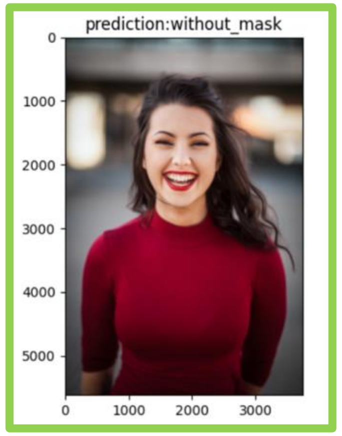


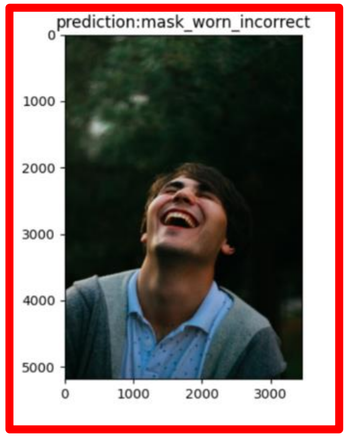


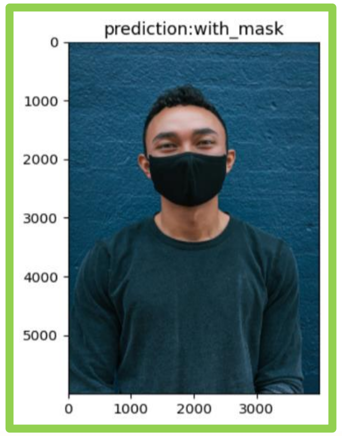
```
Test accuracy: 0.9423868312757202
```

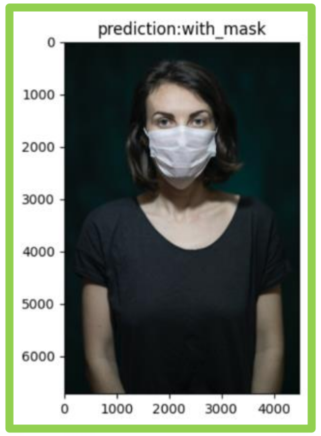
- The history of the model also looks good enough.
- The loss is going down while the accuracy is going up, which is exactly what we need.
- The learning rate remains constant at 0.001 throughout.

Results without Pre-trained Model



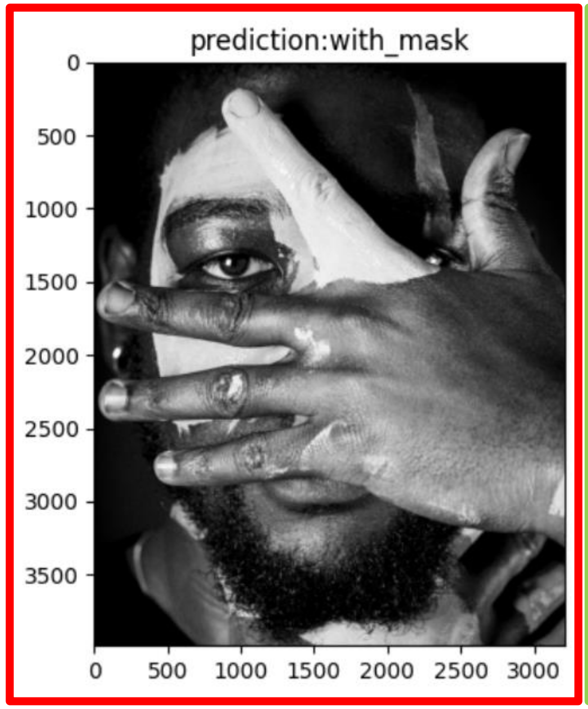


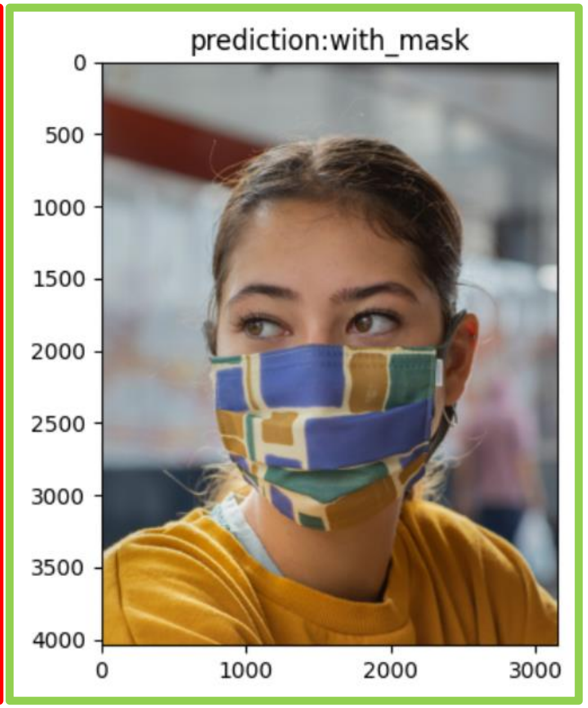


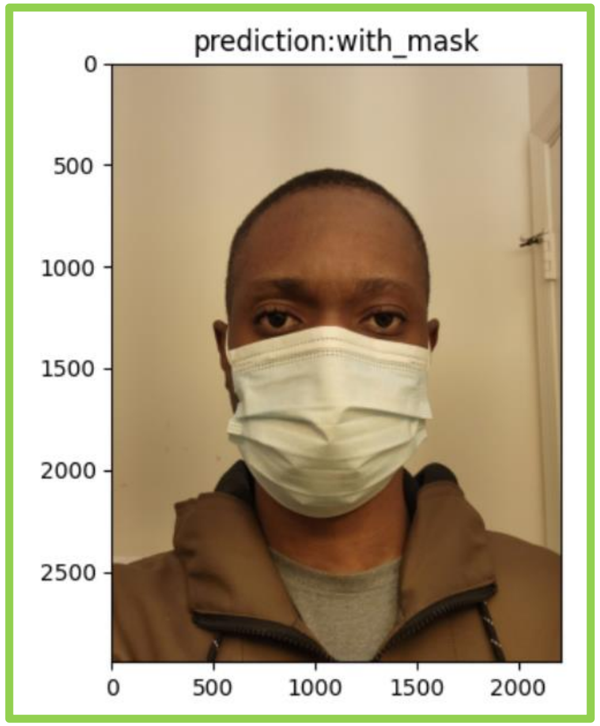


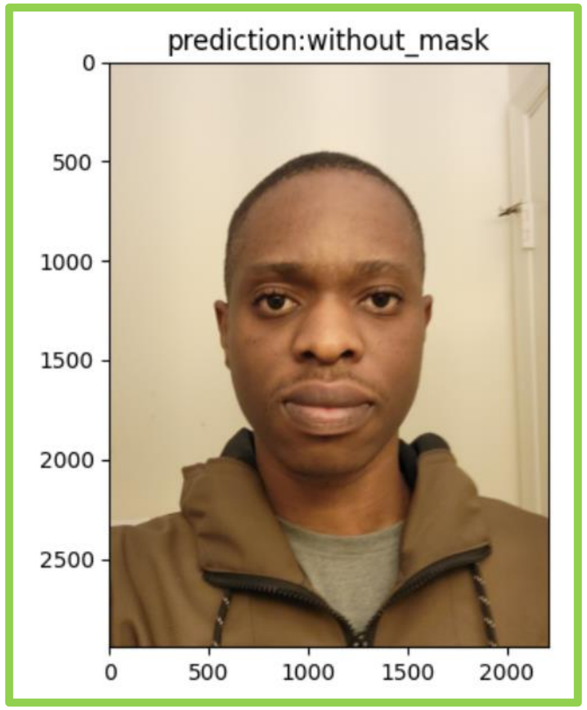
- We tested this model with the same images we tested the pre-trained model with.
- This model even though has a higher accuracy, does not predict very accurately with certain images.
- This maybe due to the imbalance in the data.

Results without Pre-trained Model

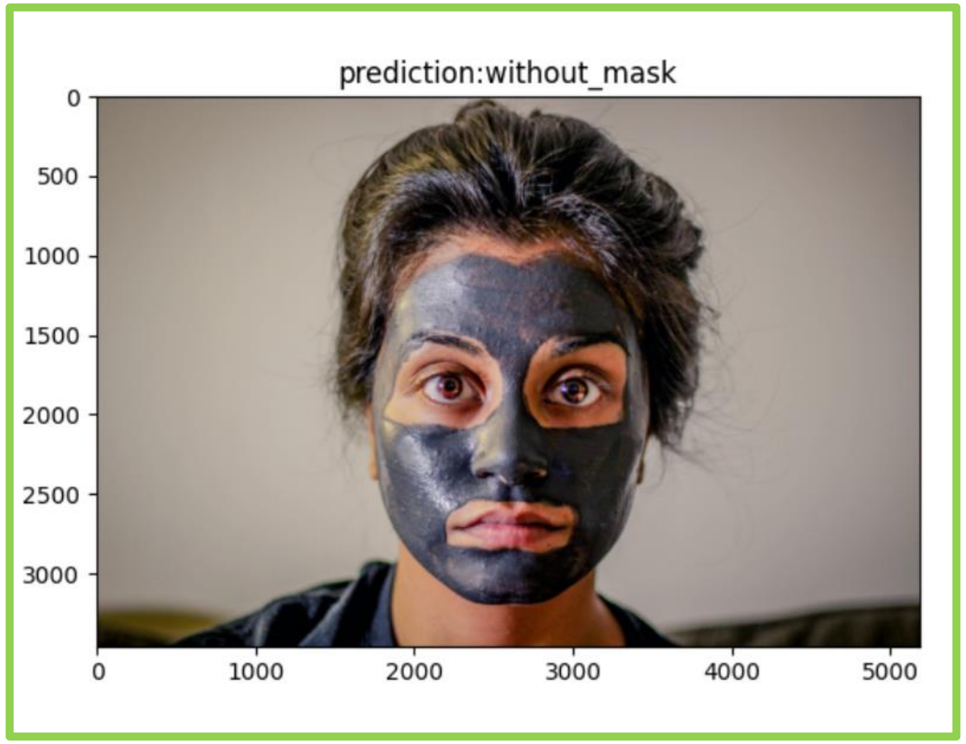


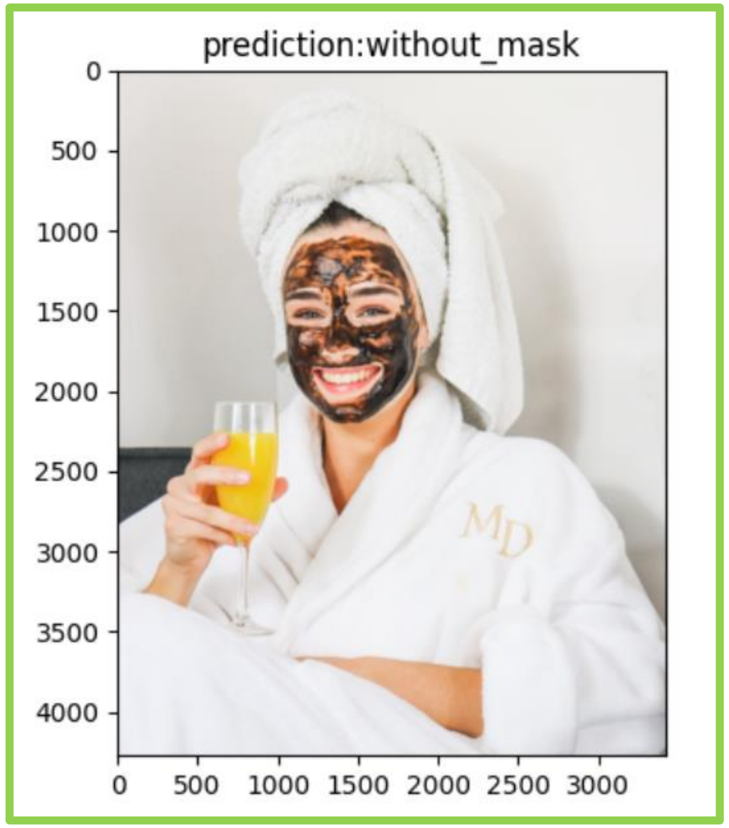






Results without Pre-trained Model





Findings

- Both models are sensitive to the face alone to predict correctly.
- The created CNN model performed better than the Xception pre-trained model.

Conclusion

- From the above results with real images, we can conclude that this model is not accurate enough to be used in the real world application.
- But more training data for the ”without_mask” and “mask_worn_incorrectly” classes will improve the model and could perform better in real world application.

Recommendations

This model could also work better with a face detection model, as the predictions are accurate when the images are focused on the face properly.
The other alternative could be, to use Faster RCNN to train the model which has the ability to create bounding boxes around the targets.

Thank You!

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC