Face Mask Detection

Multiclass Classification using Convolutional Neural Networks

By: Derya Gumustel, Amir Semsarzadeh, Umair Evans, Tarun Nappoly

Live Demo -

Agenda

- 😷 The Problem
- 😷 Methodology
- 😷 Workflow and Findings
- 😷 Roadblocks and Limitations
- 😷 Next Steps

The Problem

The Problem (cont)

Problem Statement:

Can our model reliably predict who is wearing a mask properly, improperly, or not wearing a mask at all?

The Problem

- 148 million cases | 3,127,000 deaths globally, as of Apr 26, 2021
- Plenty of people aren't wearing masks, or wearing them incorrectly, which is difficult in fighting the spread of Covid-19
- Businesses may be subject to further closures if cases increase
- As of September 2020, 97,966 businesses closed permanently in USA alone

Methodology

Data Acquisition

- 😷 Came across dataset of 70,000 images on Github from Flickr.
  - 😊 Decided on 9,000 total images for scope of our project
    - 😨 3,000 unmasked, 3,000 masked, 3,000 incorrectly masked
    - 😨 Settled for this amount due to computational constraints
- 😷 Resized images from 1024x1024 to 256x256 using PIL,OS and SYS libraries.
- 😷 This actually improved the creation of RGB arrays for each image

Unmasked faces

An example of an unmasked face in the three channel layers representing RGB.

Correctly worn masks

An example of a masked face with a correctly worn mask in the three channel layers representing RGB.

Incorrectly worn masks

Uncovered Nose

Uncovered nose and mouth

Uncovered chin

A deeper look at the Incorrectly worn masks classification

Considerations

- 😷 We grappled with the idea that our data may not be fully representative of each race/ethnicity and also different kinds of face coverings.
  - 😊 The problem being that our model may exhibit certain racial biases due to the misrepresented races in our images due to the sample limitations.
- 😷 We handpicked a few folders of photos, but don't know how truly random they were
- 😷 On the other hand, due to the robustness(image quality and quantity) of our data we needed a model that could handle whatever we threw at it and decided on using a CNN model

Workflow and Findings

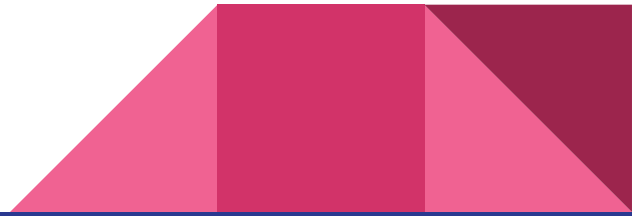
Data Transformations

- 😷 Spent less time on cleaning and significantly more time on preprocessing
  - 😊 Focused our efforts on getting the data resized to a smaller output to improve computational efficiency
  - 😊 Repeated this step for each batch of images

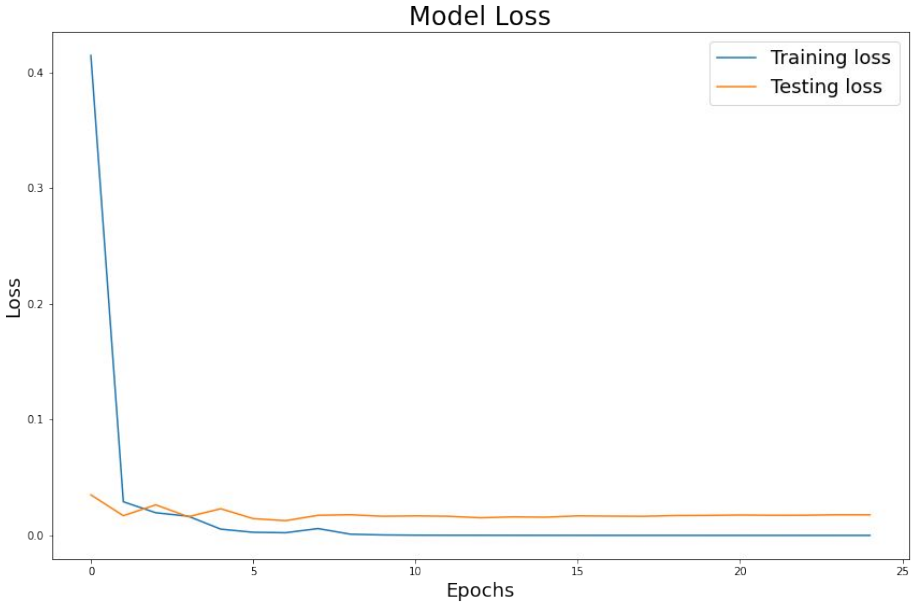
Model Structure

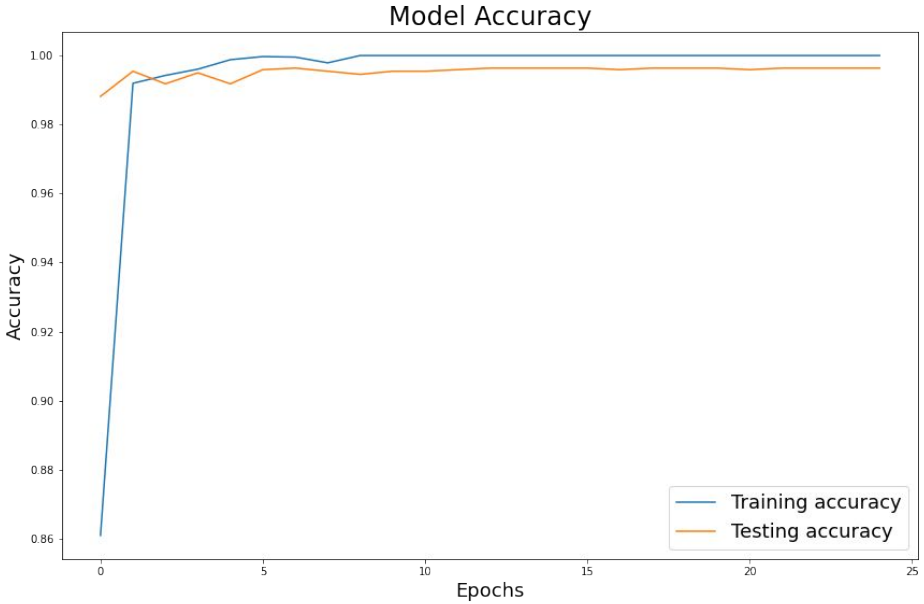
- 2 Conv2D layers with 16 nodes each
- 2 pooling layers, one following each Conv2D layer
- 1 Flatten() layer
- 1 Dense layer with 16 nodes
- Output layer with 3 outputs

When compiling this model, we used the Adam optimizer with categorical cross-entropy as the loss function

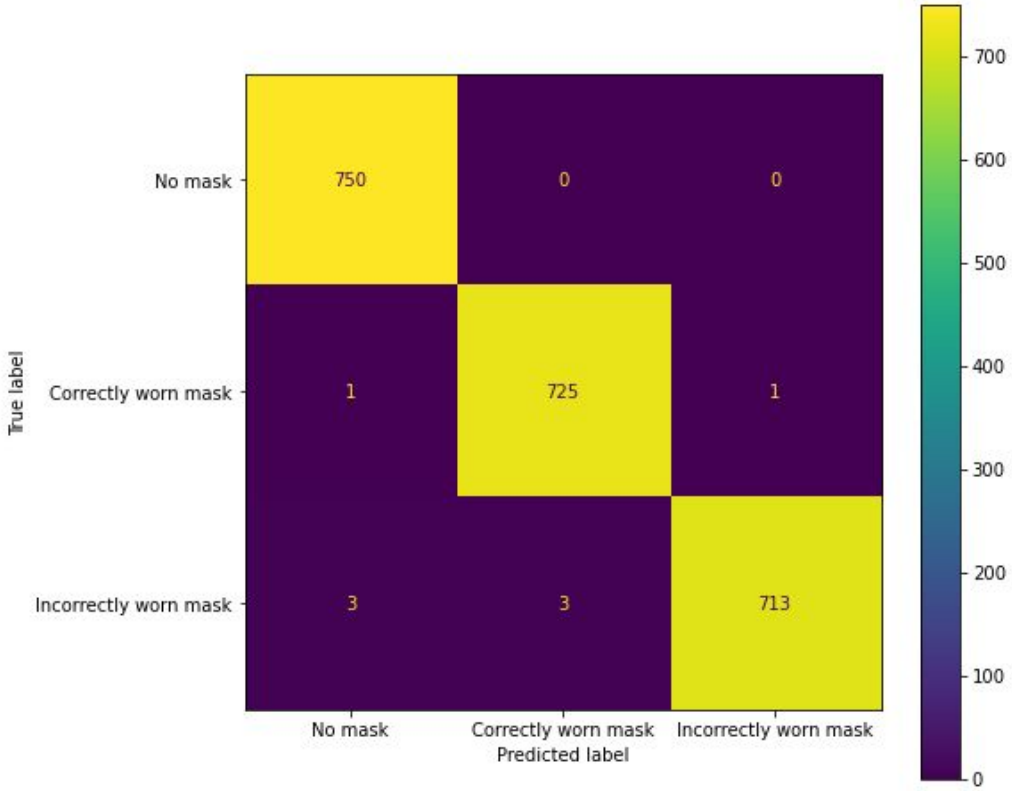


Model results and evaluation





Confusion Matrix Display



Our model made very few misclassifications!

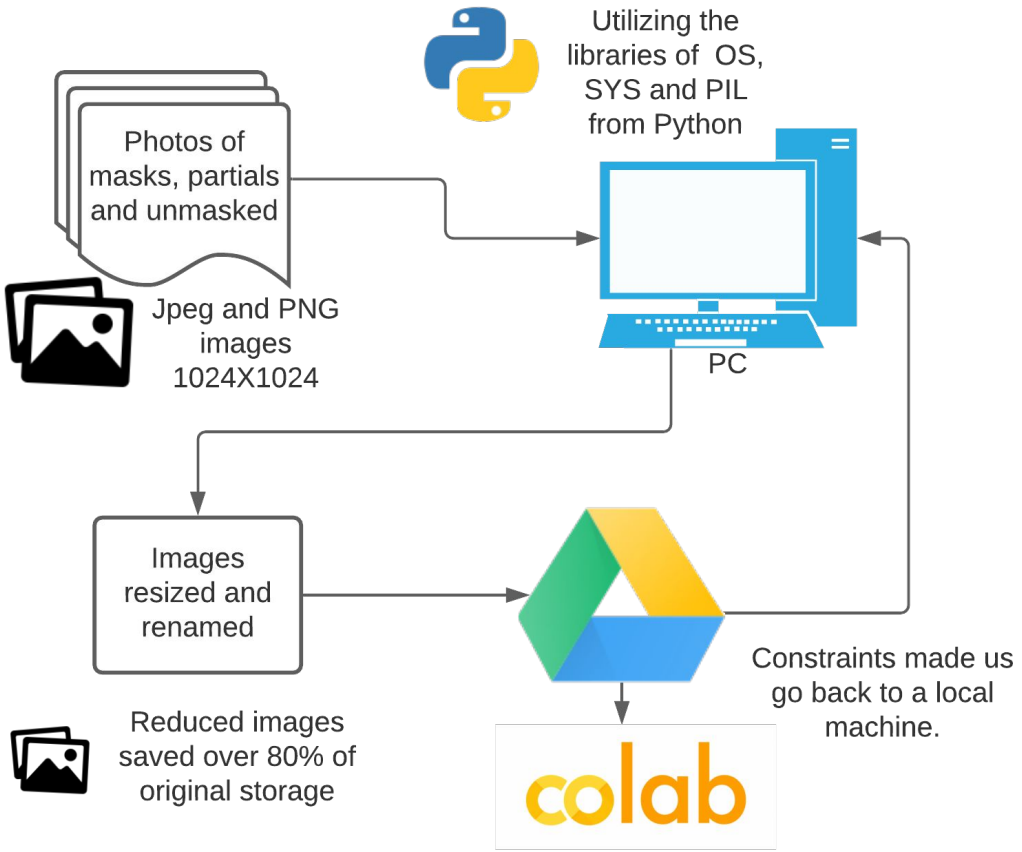
Precision and recall scores were .99 or greater for each class.

Video Streaming

- 😷 Using the opencv library we were able to put together a framework that would enable us to deploy our model in real-time using a built-in webcam
- 😷 Referenced Claire Hester's GitHub and Adrian Rosebrock's article on how to train a model to detect face masks in real-time video streams

Roadblocks, Limitations, and pain in the assets

Road Map of Project 5

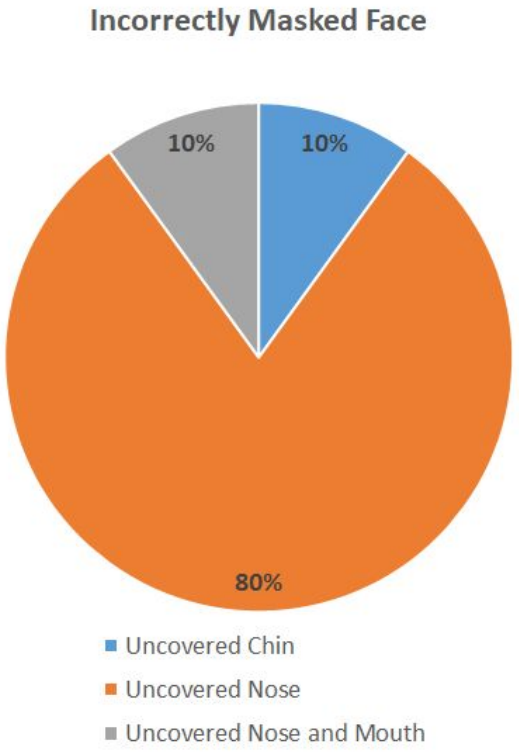


Google Colab and Google Drive

- 😷 S3 buckets were difficult and time-consuming.
- 😷 Google Colab had RAM restrictions.
- 😷 Paid solutions didn't work!

amazon S3

Imbalanced Subclass of Incorrectly Worn Masks



- 😷 Subclasses of incorrectly worn masks were very imbalanced.
- 😷 Final model is great at identifying uncovered noses, not chin.

Check Your Mask!

people wearing masks without covering their nose

Next Steps...

Next Steps

- 😷 Train model on:
  - data with multiple faces per image
  - real-images as opposed to photoshopped
  - video snippets of people
  - **Train on different types of masks**
  - **Train on subclasses of incorrectly worn masks** to make model more specific
- 😷 Deploy on Streamlit for public use

Conclusion

Conclusion

- 😷 **Mask detection is a success**, with many real-world applications
- 😷 With a **more diverse set of training data** our model can become increasingly more robust and agile in making the right classification
- 😷 **Be mindful of your computational resources** when working with computer vision
  - 😊 Think about memory, storage, and power of computing engine

Arigatou