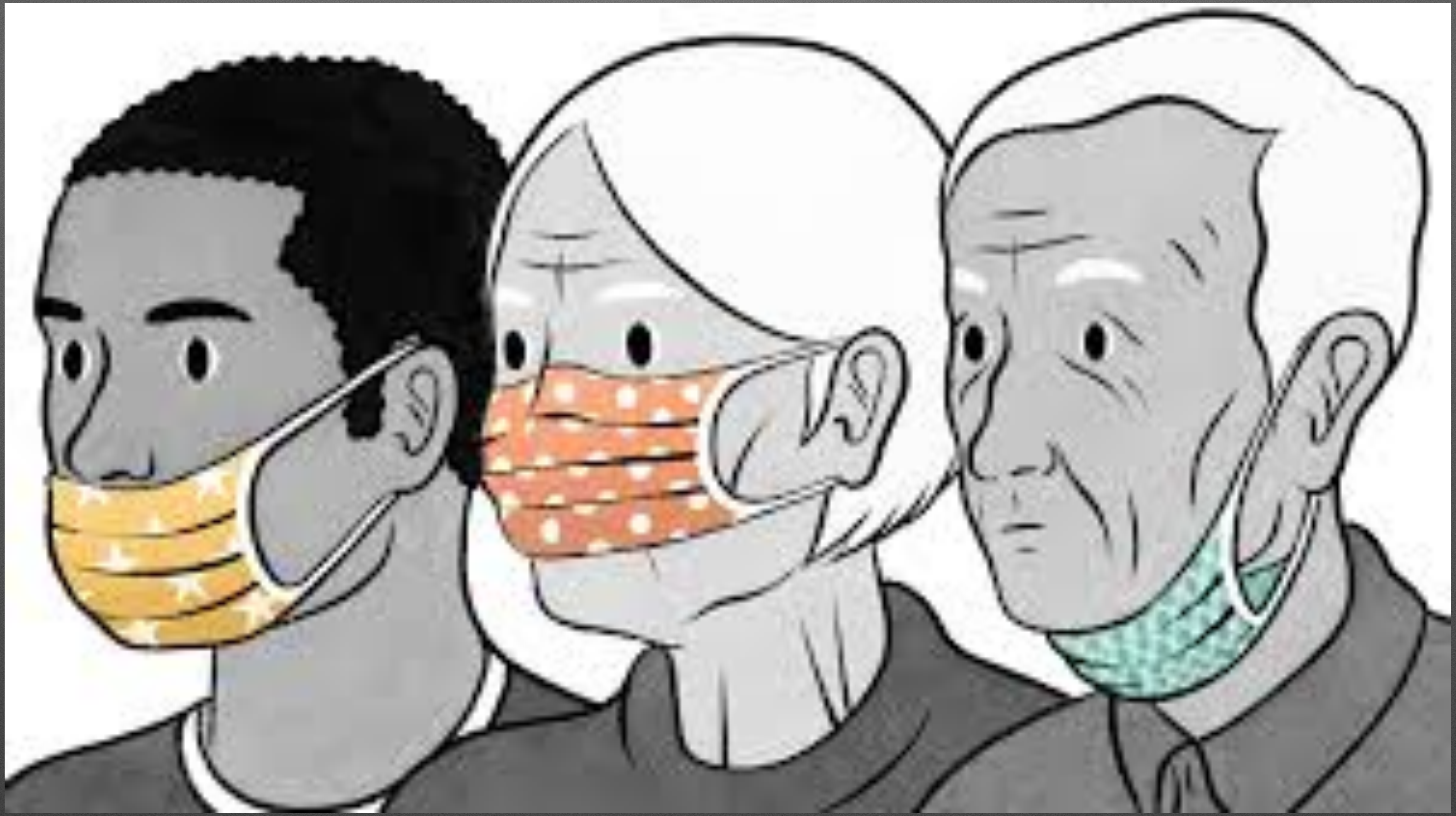


Face Mask Detection (By SSC)

Team Name:-Hugs_For_Bugs
Team Leader Name :-Aman Jain

Abstract

Research suggests that COVID-19 facemasks degrade the performance of face detection technology. The purpose of this study is to quantify this effect and identify the ideal way to train face detection models when facemasks are present. To do this, I trained a machine learning model for face detection with 1,000 regular human faces using the You Only Look Once (YOLOv4) object detection framework. I tested the face detection capabilities of the model on both regular human faces and masked faces, recording the accuracy and recall rate for each group. I then trained a new model, incorporating masked faces into the initial training dataset. The adjusted model was tested to determine if the adjusted training set imporoved performance. This research will benefit machine learning researchers and data scientists who will train and utilize facial recognition models in the midst of COVID-19 and beyond.

Introduction & Research Question

Introduction

In the COVID-19 pandemic, most people are wearing face coverings in public areas. Since face masks (obviously) cover portions of the subject's face, they interfere with face detection algorithms (Ngan et. al, 2020), which are important for biometric authentication and other security applications.

Face detection (to be distinguished from facial recognition) involves determining the presence (and location, if applicable) of human faces in an image or video. In this paper, we explore the extent to which the presence of face masks in images degrade face detection performance, and how this can be compensated for.

Research Question

How do face masks affect face detection performance in machine-learning based detection systems, and how can this be overcome?

Hypothesis

By integrating images of masked faces into the training dataset, we can produce a face detection model that is more effective in detecting masked faces.

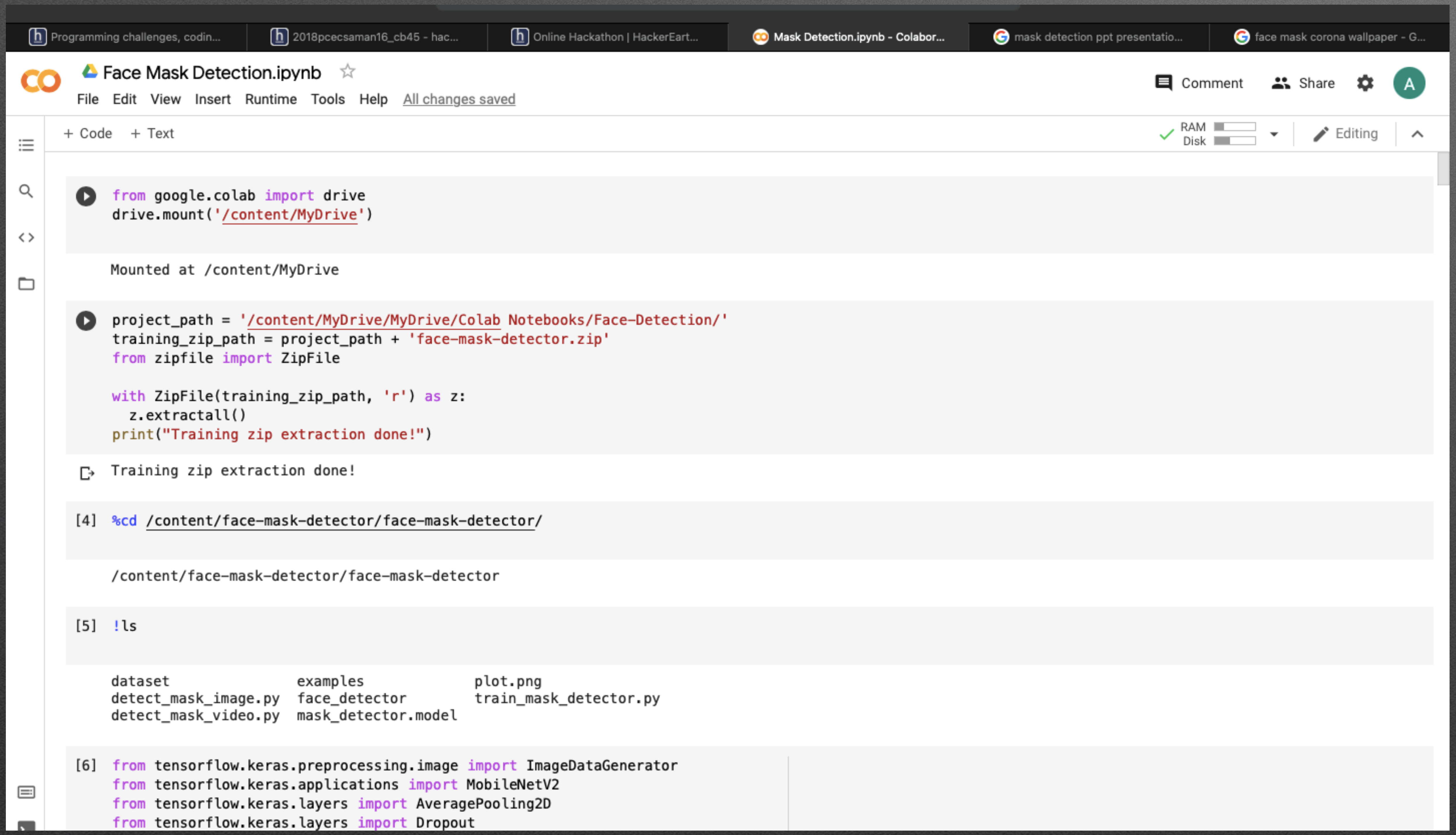
Research and Data

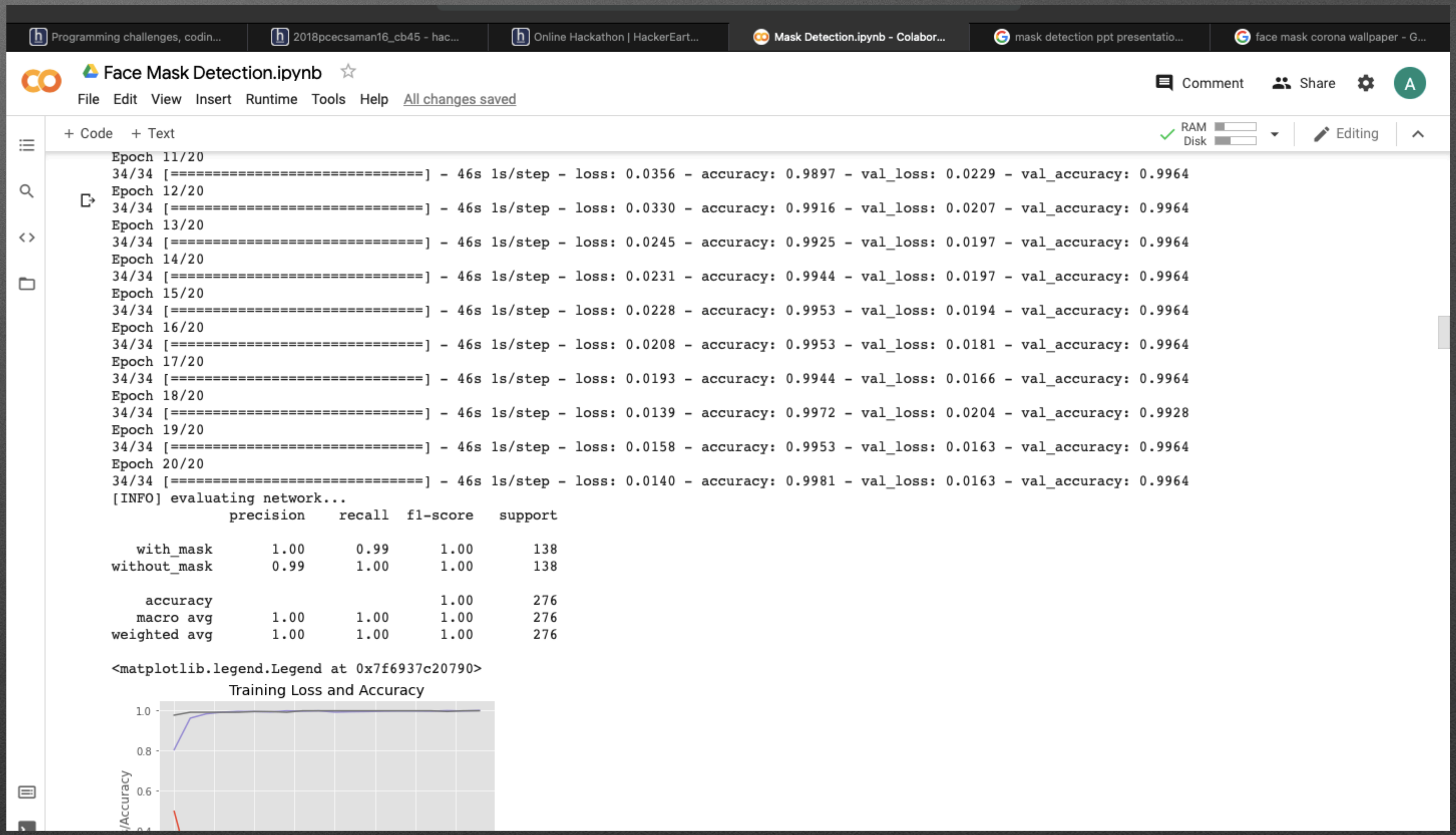
Quantitative Analysis: Trained two face detection models with different training datasets (masked vs unmasked faces) and analyzed the performance of each on both masked and unmasked faces.

Research Instruments:

- Training data acquired through Google Open Image database as well as academic sources. Manually labeled over 600 images of masked faces.
- Research conducted on personal desktop workstation: NVIDIA RTX 2060 GPU utilizing the NVIDIA CUDA toolkit.
- Trained detection models using the YOLOv4 framework by Redmon et al.

Results


Face Mask Detection.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
RAM Disk
Editing
from google.colab import drive
drive.mount('/content/MyDrive')
Mounted at /content/MyDrive
project_path = '/content/MyDrive/MyDrive/Colab Notebooks/Face-Detection/'
training_zip_path = project_path + 'face-mask-detector.zip'
from zipfile import ZipFile
with ZipFile(training_zip_path, 'r') as z:
z.extractall()
print("Training zip extraction done!")
Training zip extraction done!
[4] %cd /content/face-mask-detector/face-mask-detector/
/content/face-mask-detector/face-mask-detector
[5] !ls
dataset examples plot.png
detect_mask_image.py face_detector train_mask_detector.py
detect_mask_video.py mask_detector.model
[6] from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout


Face Mask Detection.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
+ Code + Text
RAM Disk
Editing
Epoch 11/20
34/34 [==============================] - 46s 1s/step - loss: 0.0356 - accuracy: 0.9897 - val_loss: 0.0229 - val_accuracy: 0.9964
Epoch 12/20
34/34 [==============================] - 46s 1s/step - loss: 0.0330 - accuracy: 0.9916 - val_loss: 0.0207 - val_accuracy: 0.9964
Epoch 13/20
34/34 [==============================] - 46s 1s/step - loss: 0.0245 - accuracy: 0.9925 - val_loss: 0.0197 - val_accuracy: 0.9964
Epoch 14/20
34/34 [==============================] - 46s 1s/step - loss: 0.0231 - accuracy: 0.9944 - val_loss: 0.0197 - val_accuracy: 0.9964
Epoch 15/20
34/34 [==============================] - 46s 1s/step - loss: 0.0228 - accuracy: 0.9953 - val_loss: 0.0194 - val_accuracy: 0.9964
Epoch 16/20
34/34 [==============================] - 46s 1s/step - loss: 0.0208 - accuracy: 0.9953 - val_loss: 0.0181 - val_accuracy: 0.9964
Epoch 17/20
34/34 [==============================] - 46s 1s/step - loss: 0.0193 - accuracy: 0.9944 - val_loss: 0.0166 - val_accuracy: 0.9964
Epoch 18/20
34/34 [==============================] - 46s 1s/step - loss: 0.0139 - accuracy: 0.9972 - val_loss: 0.0204 - val_accuracy: 0.9928
Epoch 19/20
34/34 [==============================] - 46s 1s/step - loss: 0.0158 - accuracy: 0.9953 - val_loss: 0.0163 - val_accuracy: 0.9964
Epoch 20/20
34/34 [==============================] - 46s 1s/step - loss: 0.0140 - accuracy: 0.9981 - val_loss: 0.0163 - val_accuracy: 0.9964
[INFO] evaluating network...
precision recall f1-score support
with_mask 1.00 0.99 1.00 138
without_mask 0.99 1.00 1.00 138
accuracy 1.00 276
macro avg 1.00 1.00 1.00 276
weighted avg 1.00 1.00 1.00 276
<matplotlib.legend.Legend at 0x7f6937c20790>
Training Loss and Accuracy
1.0
0.8
0.6


Programming challenges, codin...
2018pcecsaman16_cb45 - hac...
Online Hackathon | HackerEart...
Mask Detection.ipynb - Colabor...
mask detection ppt presentatio...
face mask corona wallpaper - G...
Face Mask Detection.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
No Mask: 91.24%
Mask: 99.98%

Programming challenges, codin... | 2018pcecsaman16_cb45 - hac... | Online Hackathon | HackerEart... | Mask Detection.ipynb - Colabor... | mask detection ppt presentatio... | face mask corona wallpaper - G...

Face Mask Detection.ipynb

File Edit View Insert Runtime Tools Help All changes saved

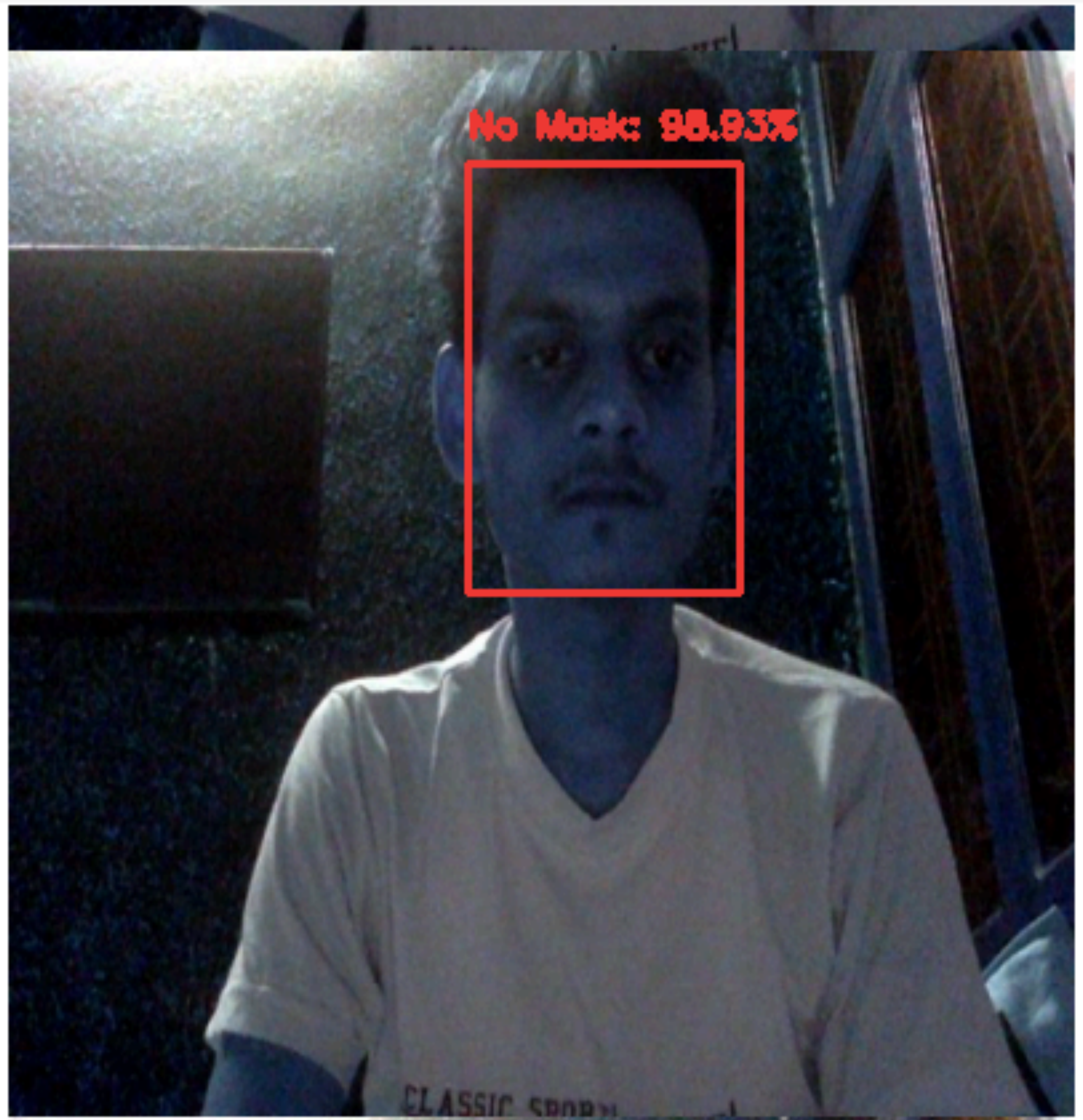
Comment Share

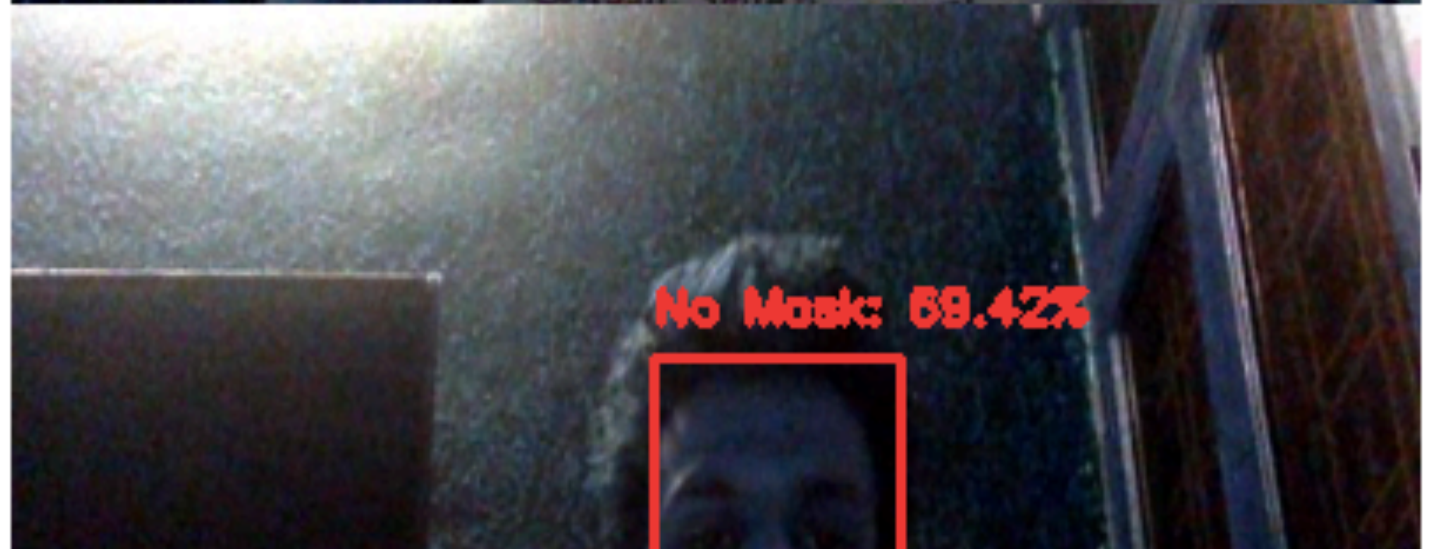
+ Code + Text

RAM Disk Editing

```
        frame=cv2.putText(frame, label, (startX, startY - 10),cv2.FONT_HERSHEY_SIMPLEX, 0.45, color, 2)
        frame=cv2.rectangle(frame, (startX, startY), (endX, endY), color, 2)

    # show the output frame
        cv2_imshow(frame)
```





Future Research

- The models in this research only detected faces -- it did not detect facemasks or distinguish masked vs. unmasked faces. Future research could focus on the applications of facemask detection.
- Although it was not the focus here, YOLO v4 is capable of making detections in real-time. More research could be conducted on real-time

Conclusion

- The results show a substantial degradation in performance in the baseline model when facemasks are present.
- The hypothesis is supported: integrating masked-face images into the training dataset increased face detection performance where facemasks are present.
- Limitations:
    - False positives-- adjusted (mask-trained) model sometimes predicts that images of facemasks (not on a person) are faces themselves. This can be fixed with hard negative examples.
    - Dataset size.

Thank You !