FACE MASK DETECTION

DSIR-824 | Nov 18, 2020

Hello!

I AM CLAIRE HESTER

Data Science Fellow at General Assembly

1 THE PROBLEM

Using computer vision, design a face mask detector that applies to both photos and live video streams

A GENTLE REMINDER...

How NOT to wear a mask

Around your neck

On your forehead

Under your nose

Only on your nose

On your chin

Dangling from one ear

On your arm

4

MODELING PROCESS

First: Build a Mask Classifier

Second: Run a face detector

Last: Assess performance on images and video

2 DATA GATHERING and EDA

853 total images …

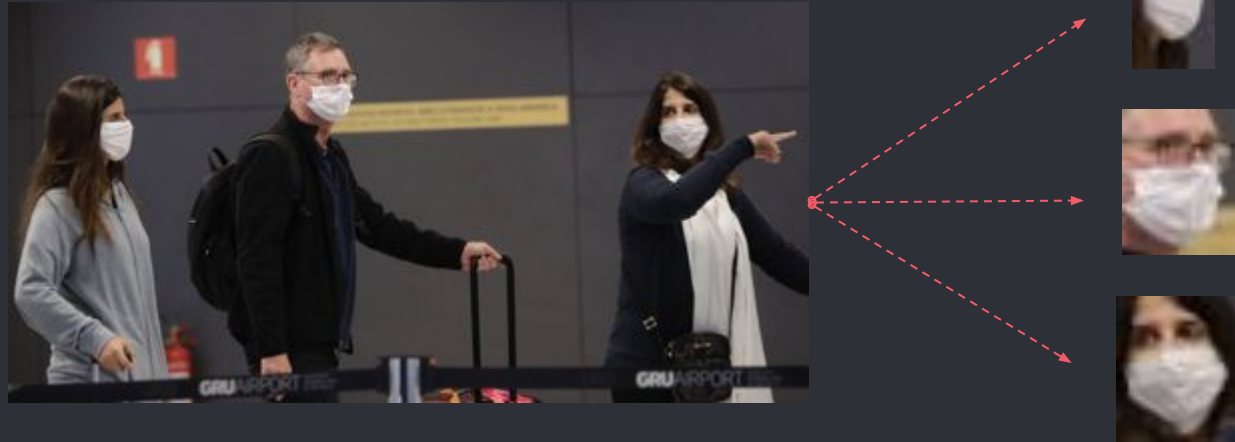- 4.8 faces per image
- Range: 1-115 faces per image

with 2,800 faces:

- 2,287 with mask worn correctly
- 418 with no mask
- 95 with mask worn incorrectly

MASK CLASSIFIER: IMAGE SEGMENTATION

3 FACE MASK CLASSIFIER

MASK CLASSIFIER: UNDER THE HOOD

| Preprocess and Augmentation | MobileNetV2 Base | Modified Head |

Preprocess and Augmentation

Encode labels and resize images
Train/test/split: 0.3 test
ImageDataGenerator random augmentations:
- Rotate
- Zoom
- Shift
- Shear
- Flip

MobileNetV2 Base

Feature Extractor
17 Bottleneck Residual Blocks
- 3 layers each
Final layer: 1x1 Convolution
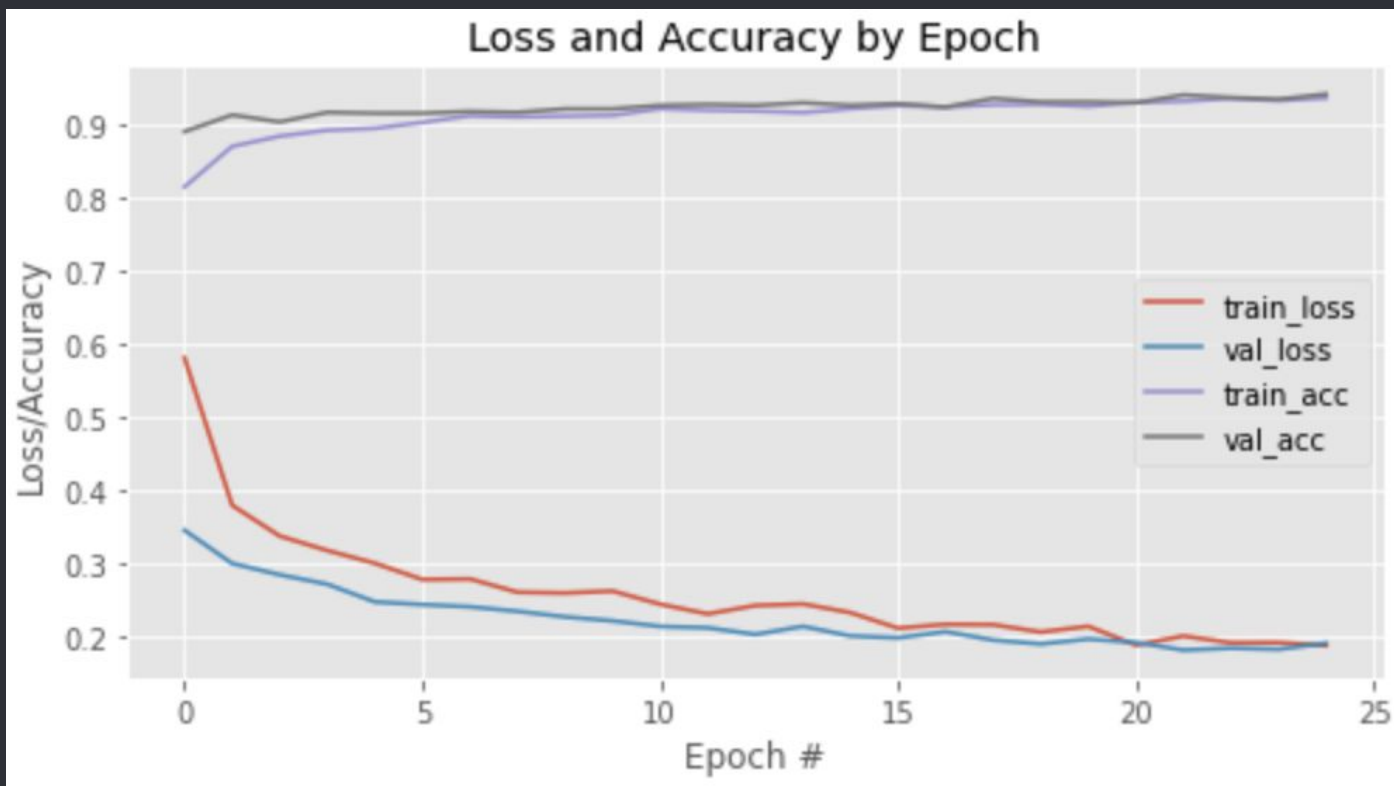Transfer learning
- ImageNet weights
Layers are frozen

Modified Head

- Average Pooling 2D
- Flatten
- Dense 64-node layer, activation: ReLU
- Dropout: 0.5
- Dense 3-node layer, activation: softmax

Validation data accuracy: 94.2%
Validation loss of 0.19

Loss and Accuracy by Epoch

CONFUSION MATRIX

PREDICTED

ACTUAL		WORN INCORRECTLY	WITH MASK	WITHOUT MASK
	WORN INCORRECTLY	6	20	3
	WITH MASK	0	680	6
	WITHOUT MASK	0	20	105

4 FACE DETECTOR

FACE DETECTION - TWO METHODS

Multi-Task Cascaded CNN (MTCNN)

Best for: images

FPS: 2.4

Total faces found: 2,670

Method: three-stage cascaded CNN:

- P-Net
- R-Net
- O-Net

OpenCV Built-In DNN

Best for: video

FPS: 10.8

Total faces found: 1,662

Method: SSD with Resnet-10 backbone

5 BRINGING IT ALL TOGETHER

SO, HOW DID WE DO?

- - 4 faces detected
- - Correct classifications

- - 14 faces detected
- - Mostly correct classifications

SO, HOW DID WE DO?

- 3 faces detected - this is correct!
- 2 correct classifications

- 14 faces present
- 3 faces detected with correct classifications

DEMO

Lets see how this performs in real time!

6 WHAT'S NEXT?

Face Detection with Mask

One obvious challenge: face masks naturally cover the face, making it challenging to detect faces in the first place.

Benefits of a Two-Step Model

Allows for greater fine-tuning

Suggested applications

Track use of face masks in public spaces

Connected to hardware, can prevent non-masked individuals from entering buildings such as retail stores, restaurants, etc.

THANK YOU!

You can find me at:
LinkedIn Github

APPENDIX

SOURCES

https://www.kaggle.com/andrewmvd/face-mask-detection

https://app.roboflow.com/dataset/capstone-face-mask/2

https://arxiv.org/pdf/2011.02371v1.pdf

https://www.pyimagesearch.com/2019/07/08/keras-imagedatagenerator-and-data-augmentation/

https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/about-face-coverings.html

https://medium.com/@iselagradilla94/multi-task-cascaded-convolutional-networks-mtcnn-for-face-detection-and-facial-landmark-alignment-7c21e8007923

https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/

https://arxiv.org/pdf/1801.04381.pdf

https://github.com/opencv/opencv/blob/master/modules/dnn/misc/face_detector_accuracy.py

https://docs.python.org/3/library/xml.etree.elementtree.html

https://www.pyimagesearch.com/2019/07/08/keras-imagedatagenerator-and-data-augmentation/

https://www.kaggle.com/notadithyabhat/face-mask-detector

MOBILENETV2 - A DEEPER DIVE

Input	Operator	t	c	n	s
$224^2 \times 3$	conv2d	-	32	1	2
$112^2 \times 32$	bottleneck	1	16	1	1
$112^2 \times 16$	bottleneck	6	24	2	2
$56^2 \times 24$	bottleneck	6	32	3	2
$28^2 \times 32$	bottleneck	6	64	4	2
$14^2 \times 64$	bottleneck	6	96	3	1
$14^2 \times 96$	bottleneck	6	160	3	2
$7^2 \times 160$	bottleneck	6	320	1	1
$7^2 \times 320$	conv2d 1x1	-	1280	1	1
$7^2 \times 1280$	avgpool 7x7	-	-	1	-
$1 \times 1 \times 1280$	conv2d 1x1	-	k	-	

Block 1
Block 2

⋮

Block 15
Block 16
Block 17
1x1 Convolution

1x1 Expansion Layer
BatchNorm
ReLu6
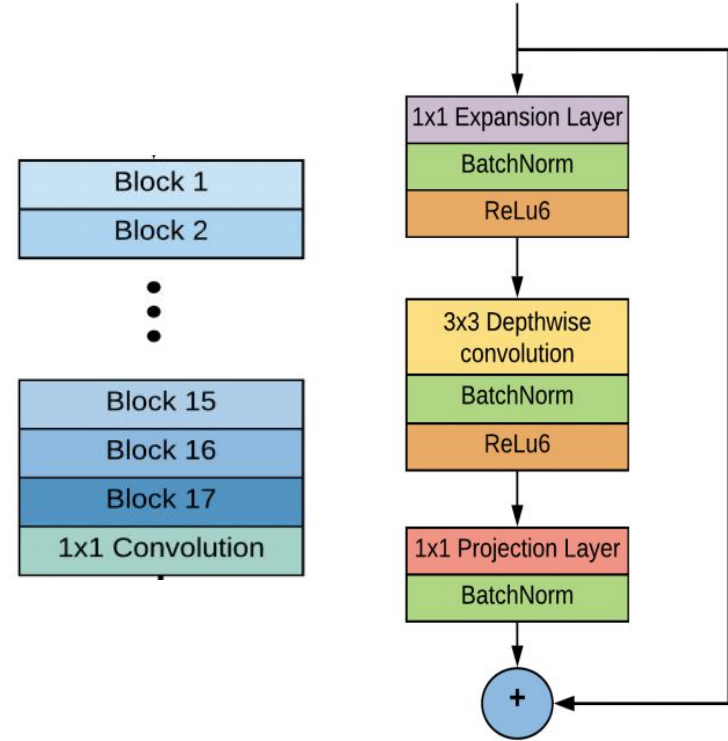
3x3 Depthwise convolution
BatchNorm
ReLu6

1x1 Projection Layer
BatchNorm

+

Fig. 2: Bottleneck Residual block

Image source: https://arxiv.org/pdf/2011.02371v1.pdf