COMP6721 Applied Artificial Intelligence (winter 2022)
Project Assignment,

AI Face Mask Detector

Team Member Information

Project Team Name:

NS_18

Team Members:

Ali Eghmazi (**40224452**) - Data Specialist

Damanpreet Arora (**40201882**) - Training Specialist

Jamal Habibi Markani (**40224697**) - Evaluation Specialist

Outline:

AI Face Mask Detector is an Artificial Intelligent application which is based on CNN model and can analyze face images and classify it whether a person is wearing a mask or not and identify the type of mask which is being worn by the person. This application is developed in two stages.

- In the initial phase, various category of image datasets including of a person without mask , with mask are created. Afterward a appropriate Convolution Neural Network(CNN) architecture model is implemented and The model is trained with training datasets to improve its accuracy and performance. Then the trained model has been used for evaluation under examination of test dataset, The performance has been computed for accuracy and F1 score for test dataset.
- In the second stage, we aimed to extract potential Bias of the trained model Bias two feature of gender (male/female) and age of the candidate image and, do analysis of reason of bias and apply possible action in order to remove bias for mentioned features. Next, the K-fold cross validation is applied. The implemented K-fold cross validation is being applied on the retrained model with bias and previous model. Evaluation metrics are being calculated and compared for the both cases and also give insights regarding the system's performance

1- Resources for the dataset:

As the size of the dataset is more than 250MB, so it is loaded in google drive.

Link to Dataset in google drive:
https://drive.google.com/drive/folders/13T7bS-2lJ-f6Ua1Qo9Dm580YTJqlJTN2

2- Dataset

The purpose of providing a dataset is to test and train a deep learning model (Convolutional Neural Network) for recognizing whether someone is wearing a mask or not, and what kind of mask they are wearing. The dataset covers 5 types of images: without a mask, with cloth_mask, with surgical_masks, with N95_masks, and Valve_mask. Approximately an average of 400 images for each class and overall there are 2000 images for without masks class. We have created our Dataset by collecting them from multiple sources and combining them to create an appropriate dataset. The current submission has less data as compared to the original dataset, the size for the zip file exceeded 250MB (373 MB). The original dataset can be found at:

https://drive.google.com/drive/folders/1E4q-J6yXMugG00LYEyuhqRtIV9tSCvu6

References:

just provided all the majority of the images for our dataset.it has people with and without masks and with different categories of masks.

This dataset has more than 6k images and was published as an open-access dataset to fight against COVID-19. The name of the provider is "**Humans in the Loop**". there is a form to fill out to access the datasets.

- https://humansintheloop.org/resources/datasets/medical-mask-dataset/

Below datasets are provided by kaggle https://www.kaggle.com under Database Contents License. kaggle offers a public data platform for Artificial Intelligence education and etc.

These datasets are for mask detection purposes.

- https://www.kaggle.com/datasets/omkargurav/face-mask-dataset
- https://www.kaggle.com/datasets/mloey1/medical-face-mask-detection-dataset

The Images for our project were manually categorized into five folders or five classes.

1. cloth_mask
2. surgical_mask
3. with_N95_mask
4. with_N95_mask_valve
5. without_mask

The updated dataset has been extracted from below references:

- https://www.kaggle.com/datasets/sumansid/facemask-dataset?resource=download-directory

The below Dataset is open source for any deep learning activities. It contains 7971 images. The dataset is composed of WIDER Face and MAFA, it can download from below link,

- https://github.com/AIZOOTech/FaceMaskDetection

3- Convolutional Neural Network Architecture

1) Dataset Extraction:

Downloading the dataset from the google drive and unzipping dataset will look as below:

- cloth_mask
- surgical_mask
- with_N95_mask
- with_N95_mask_valve
- without_mask

DatasetFrames is created by iterating all images recursively in every folder and labeling them according to the below categories:

0: **without mask**

1: with_surgical mask,

2: with_cloth mask,

3: with_N95_ mask.

4: valve Mask

3-1 Output DatasetFrame:

0.0	582
1.0	432
2.0	371
3.0	323
4.0	292

Index	Category	Count of Images	Updated count of Images
0	'without_mask'	582	643
1	'surgical_mask'	432	550
2	'cloth_mask'	371	477
3	'N95_mask'	323	346
4	'N95_mask_valve'	292	316

3-2 Dataset class:

panda's dataframe, MaskDetectionDataset classes are deployed to query images via pytorch. Following action has been done on all images:

- Transforming them by using torchvision.transforms function
- resizing them to 32*32,(1st phase)
- resizing them to 128*128,(2nd phase)
- converting to torch Tensor
- normalizing them with standard mean and std values which then forces the network values to be between 0-1.

3-3 Defining CNN model:

In the model, nn.module has been deployed in order to build neural network models.

The part 1 model contains only 3 layers of CNN, while the newly develop model layer has been changed to 4 layers.

Old model parameters as below:

Index	Blocks	Activation Dimension	Activation Size	# Param
1	Input layer	(32, 32, 3)	3072	0
2	CONV1(K=3,S=1: outchannel=12)	(32, 32,12)	12288	336
3	BatchNorm2d	(32,32,12)	12288	24
4	CONV2(K=3,S=1: outchannel=20)	(32, 32,20)	20480	2180

5	BatchNorm2d	(32, 32,20)	20480	40
6	MaxPool2d	(16,16,20)	5120	
7	CONV2(K=3,S=1: outchannel=32)	(16,16,32)	8192	5792

The New model Param which has 4 layer according to detail information which is mentioned in below:

Train model dimension and parameter:

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1         [-1, 32, 128, 128]             896
       BatchNorm2d-2         [-1, 32, 128, 128]              64
         MaxPool2d-3           [-1, 32, 64, 64]               0
            Conv2d-4           [-1, 32, 64, 64]           9,248
       BatchNorm2d-5           [-1, 32, 64, 64]              64
            Conv2d-6           [-1, 64, 64, 64]          18,496
       BatchNorm2d-7           [-1, 64, 64, 64]             128
         MaxPool2d-8           [-1, 64, 32, 32]               0
            Conv2d-9           [-1, 64, 32, 32]          36,928
      BatchNorm2d-10           [-1, 64, 32, 32]             128
        MaxPool2d-11           [-1, 64, 16, 16]               0
           Linear-12                    [-1, 5]          81,925
================================================================
Total params: 147,877
Trainable params: 147,877
```

In __init__() we define the layers and subsequent dimensions within them for the CNN as follows :

- 4 convolution layers (for channels):
  - **Conv1** applies Conv2d with input channels as 3(for RGB image) and output channels as 32.
  - A kernel size of 3 and padding of 1 result in the same dimensions for 128*128 images. `# (128 - 3 + 1 + 2*1)=128`
  - **BatchNorm2d** is used to accelerate the training process
  - **maxpool2d** is used to resize parameters and to accelerate the training process

- ➢ The same is true for **conv2**, **conv3, conv4** with 16 input channels and 8 output channels.
- 4 fully connected layers (for features):
  - ➢ fc1 applies linear with the input of (64*16*16) to output features as 5.

Moreover, we carry forward pass in the forward() method, in which we bypass the data into computation graphs (i.e. neural networks), which map inputs of Tensor to predictions, as follows in current CNN algorithms:

- ➢ Initial out calls torch.nn.functional's **max_pool2d** and evaluates conv1 from the convolution layer with a activation function with a stride of 2, thereby reducing the overall size of the current image by half.
- ➢ Further computing output for a **rectified linear unit** (ReLU) activation function.
- ➢ The conv2 action was carried forward, resulting in the image dimensions being updated:
- ➢ Activate the above features element-by-element using tanh.
- ➢ Finally passing to the output layer and returning the output.

3-4 Preparation of Dataset by splitting:.

In the First phase, there is a 75/25 split between training and testing data. This can be achieved by using the train_test_split() function, which takes a dataset label as a parameter. To train the model, the data is loaded using Dataloader, using it in batches of 32 at a time. Therefore, there are 500 samples for testing out of 2000 images. While in the second phase of the project data has been split into 80/20. Kindly note that dataset has been increase in 2nd part in order to tackle Bias and performance. In both case, splitting has been done randomly.

3-5 Training the Model:

We use “Adam” to optimize its parameters and the learning rate is set to 0.001. Our next step is to iterate through the training dataset and retrieve batches to get images and labels, pass them to the model, and compute the loss. For this project, the epoch number is 20, but you can change it.

It is well worth to note that in the 1st step of project image size has been transformed to 32 * 32 while in the second step it was transformed to 128* 128.

3-6 Saving & loading the Trained Model

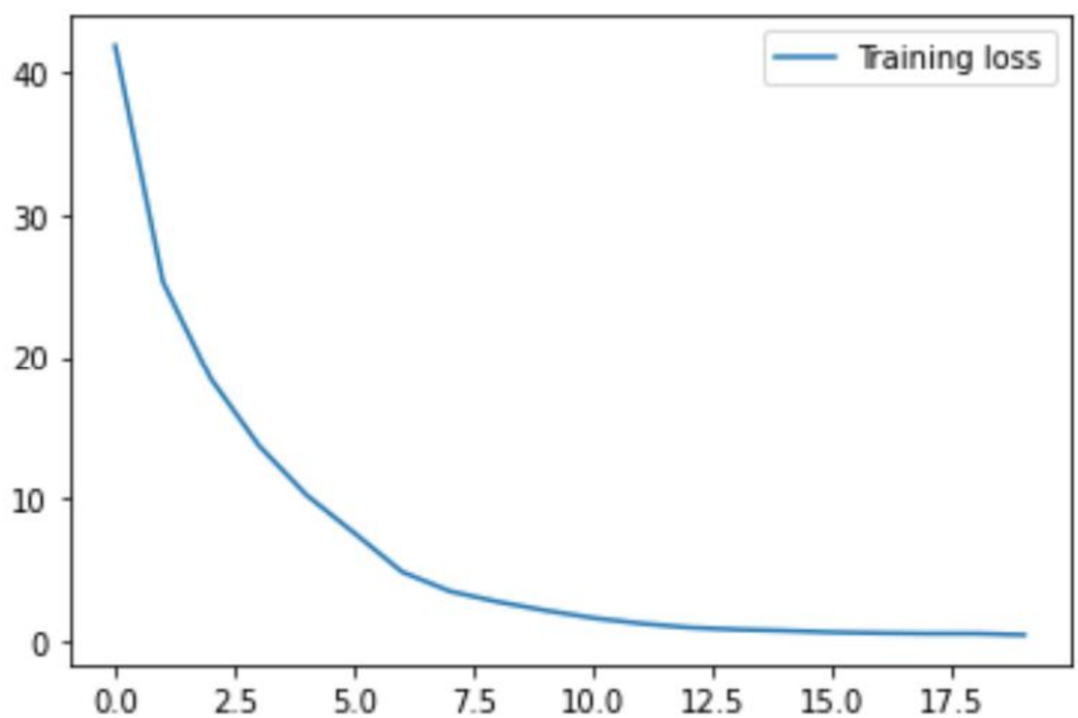
Model has been saved into the .pth file in order to save the time of training and it can be loaded into the evaluation model in order to do the evaluation. The model has been saved for various stage of project. Training, Bias of gender, Bias of age, Kfold and trained model.

4 Evaluate the model:

We Use matplotlib to predict labels on the X-axis and the actual labels on the Y-axis. The first step is to create empty tensors each for prediction and actual data, iterate through validate dataset and retrieve batches to get images and labels to evaluate the accuracy on unseen images. Afterward, the sequence of tensors are concatenate in the given size. Then, we define five classes (0 : without mask, 1 with_surgical mask, 2 with_cloth mask, 3 with_N95_ mask. 4: valve Mask) and plot confusion matrix having actuals and predictions on its axis.

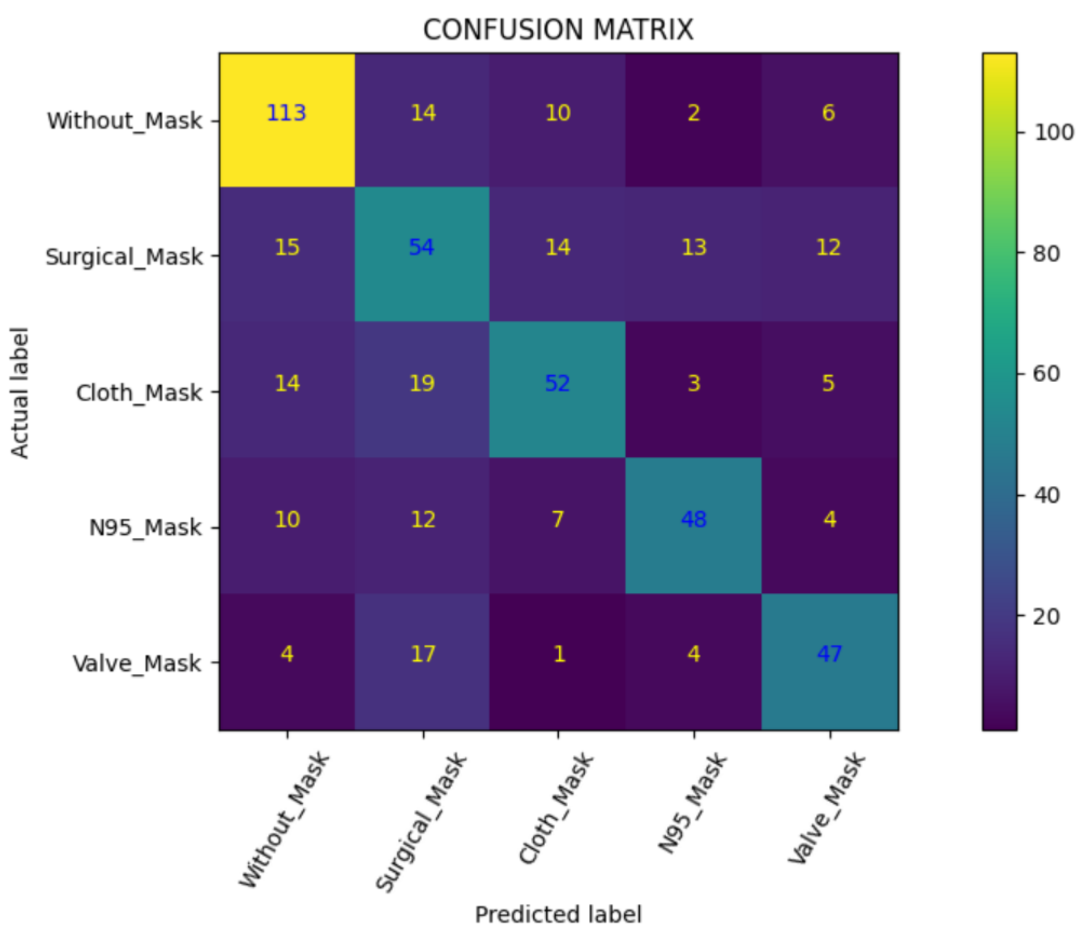
4-1 Trained model loss trend and output of Confusion Matrix:

According to CNN model, the loss of model along with epochs need to be reduced. As following trend demonstrates, the same trend has been observed in the training phase, by increasing the epoch numbers, the loss convergence will be come less as expected.



1st phase project result: Learning_rate= 0.001, epochs = 20

	precision	recall	f1-score	support
Without_Mask	0.72	0.78	0.75	145
Surgical_Mask	0.47	0.50	0.48	108
Cloth_Mask	0.62	0.56	0.59	93
N95_Mask	0.69	0.59	0.64	81
Valve_Mask	0.64	0.64	0.64	73
accuracy			0.63	500
macro avg	0.63	0.61	0.62	500
weighted avg	0.63	0.63	0.63	500



4-2 Prediction of images using trained Model:

It is used for prediction of any new image and categorized according to the 5 categories which have been trained. The output will be actual and predicted values.

4-4 Bias detection:

Various possible Bias could be existed in the trained model for instance Age, Gender and Race. The biased model directly affects the model performance and accuracy. Therefore, related action and comprehensive analysis need to be conducted in order to eliminate the existing biases. Following major action need to be done for this objective:

- ➢ Extracted Bias for following categories:
    - ▪ Gender (Female and Male)
    - ▪ With age (Young and Adult)
- ➢ Analysis the reason of bias
- ➢ Possible action for Bias remedy

Following table is for male and female bias.

- Training is done 80% of each gender datasets (it is selected randomly)

Table1: Male and female precision, recall, F1-score for every category

	Male			Female		
	precision	recall	f1-score	precision	recall	f1-score
Without_Mask	84%	72%	78%	67%	76%	71%
Surgical_Mask	63%	72%	68%	80%	61%	69%
Cloth_Mask	46%	53%	49%	56%	60%	58%
N95_Mask	53%	63%	58%	71%	69%	70%
Valve_Mask	58%	33%	42%	65%	71%	68%
Total Average	63%			67%		

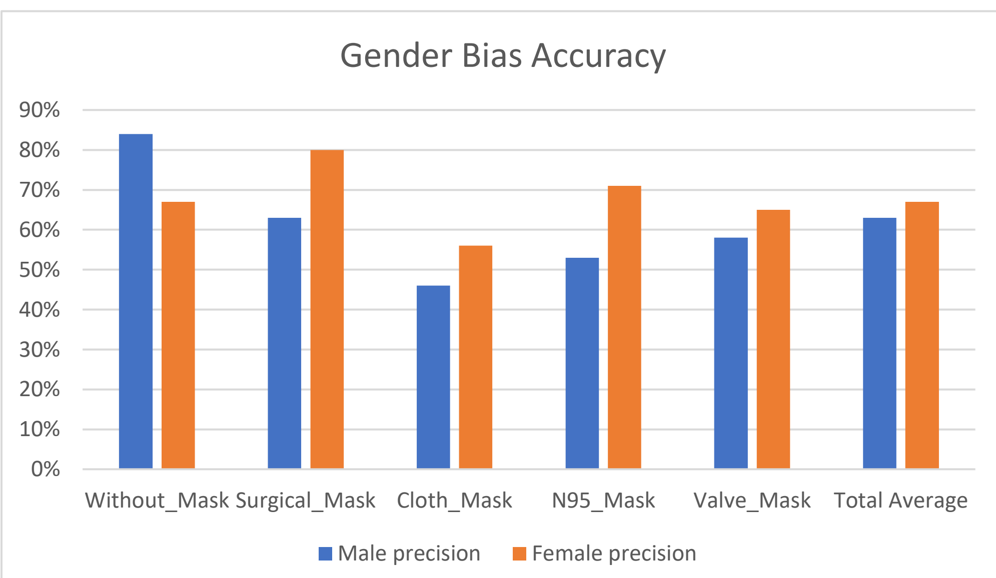
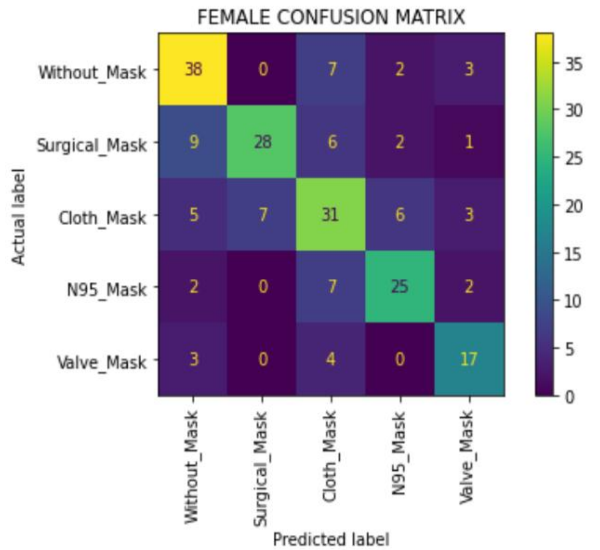


Figure : Gender bias accuracy distribution for individual categories

```
female_bias
               precision    recall  f1-score   support

 Without_Mask       0.67      0.76      0.71        50
Surgical_Mask       0.80      0.61      0.69        46
   Cloth_Mask       0.56      0.60      0.58        52
     N95_Mask       0.71      0.69      0.70        36
   Valve_Mask       0.65      0.71      0.68        24

     accuracy                           0.67       208
    macro avg       0.68      0.67      0.67       208
 weighted avg       0.68      0.67      0.67       208

[[38  0  7  2  3]
 [ 9 28  6  2  1]
 [ 5  7 31  6  3]
 [ 2  0  7 25  2]
 [ 3  0  4  0 17]]
```




```
Male_bias
               precision    recall  f1-score   support

 Without_Mask       0.84      0.72      0.78        65
Surgical_Mask       0.63      0.72      0.68        36
   Cloth_Mask       0.46      0.53      0.49        43
     N95_Mask       0.53      0.63      0.58        30
   Valve_Mask       0.58      0.33      0.42        21

     accuracy                           0.63       195
    macro avg       0.61      0.59      0.59       195
 weighted avg       0.64      0.63      0.63       195

[[47  5  9  3  1]
 [ 3 26  5  2  0]
 [ 3  7 23  7  3]
 [ 0  2  8 19  1]
 [ 3  1  5  5  7]]
```

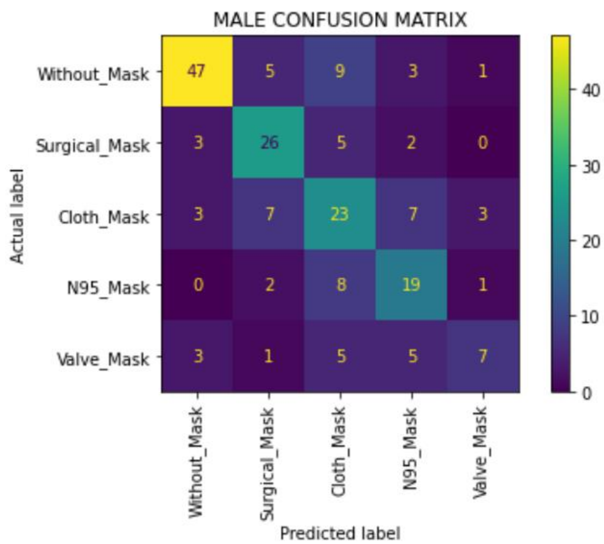


Figure : Gender bias test results and confusion matrix

Analysis:

As per above snapshots and tables, our dataset was bias towards female when we predicted each gender-based test data. Overall average of female accuracy is 4% better than male one. Therefore, based on confusion matrix outcome, following major category need to be improved:

I. Surgical mask (male)
II. Cloth Mask (male)
III. N95 Mask (male)
IV. N95 Valve Mask (male)
V. Without mask (female)

In the meantime, it is wise to address low performance of With_out mask category for the female. Thus, to overcome the gender-based bias, following 184 photos for each category has been added in order to address above mention issue.

	Male	Female
Without_Mask		71
Surgical_Mask	20	
Cloth_Mask	39	
N95_Mask	22	
Valve_Mask	32	

Below Is the result of action for Bias elimination:

Table2: Male and female precision, recall, F1-score for every category

	Male			Female		
	precision	recall	f1-score	precision	recall	f1-score
Without_Mask	80%	75%	78%	73%	80%	76%
Surgical_Mask	54%	49%	51%	69%	59%	64%
Cloth_Mask	59%	64%	61%	51%	48%	50%
N95_Mask	58%	82%	68%	58%	72%	64%
Valve_Mask	56%	36%	43%	74%	58%	65%
Total Average	64%			64%		

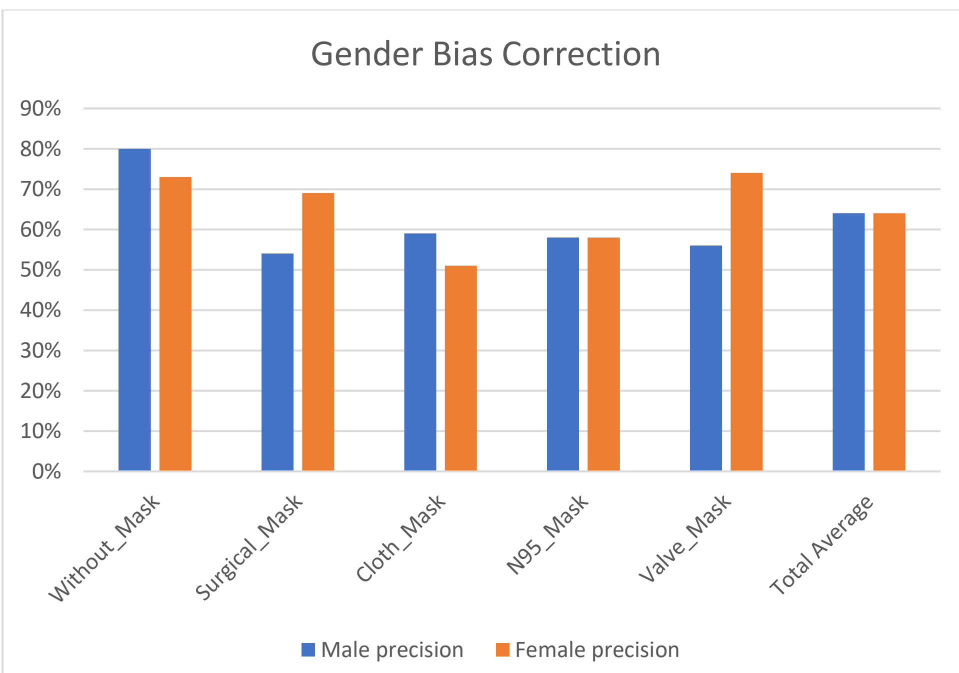
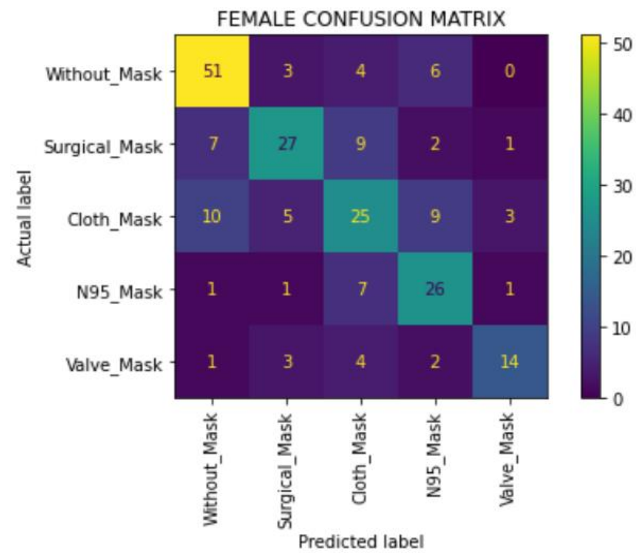


Figure : Gender bias accuracy distribution for individual categories

```
female_bias
               precision    recall  f1-score   support

 Without_Mask       0.73      0.80      0.76        64
Surgical_Mask       0.69      0.59      0.64        46
   Cloth_Mask       0.51      0.48      0.50        52
     N95_Mask       0.58      0.72      0.64        36
   Valve_Mask       0.74      0.58      0.65        24

     accuracy                           0.64       222
    macro avg       0.65      0.63      0.64       222
 weighted avg       0.65      0.64      0.64       222

[[51  3  4  6  0]
 [ 7 27  9  2  1]
 [10  5 25  9  3]
 [ 1  1  7 26  1]
 [ 1  3  4  2 14]]
```




```
Male_bias
               precision    recall  f1-score   support

 Without_Mask       0.80      0.75      0.78        65
Surgical_Mask       0.54      0.49      0.51        43
   Cloth_Mask       0.59      0.64      0.61        47
     N95_Mask       0.58      0.82      0.68        34
   Valve_Mask       0.56      0.36      0.43        28

     accuracy                           0.64       217
    macro avg       0.61      0.61      0.60       217
 weighted avg       0.64      0.64      0.63       217

[[49  5  6  3  2]
 [ 9 21  7  5  1]
 [ 3  6 30  4  4]
 [ 0  2  3 28  1]
 [ 0  5  5  8 10]]
```

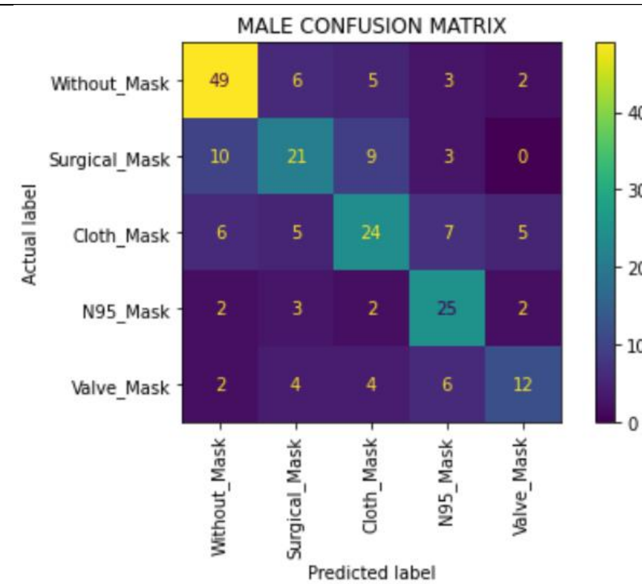
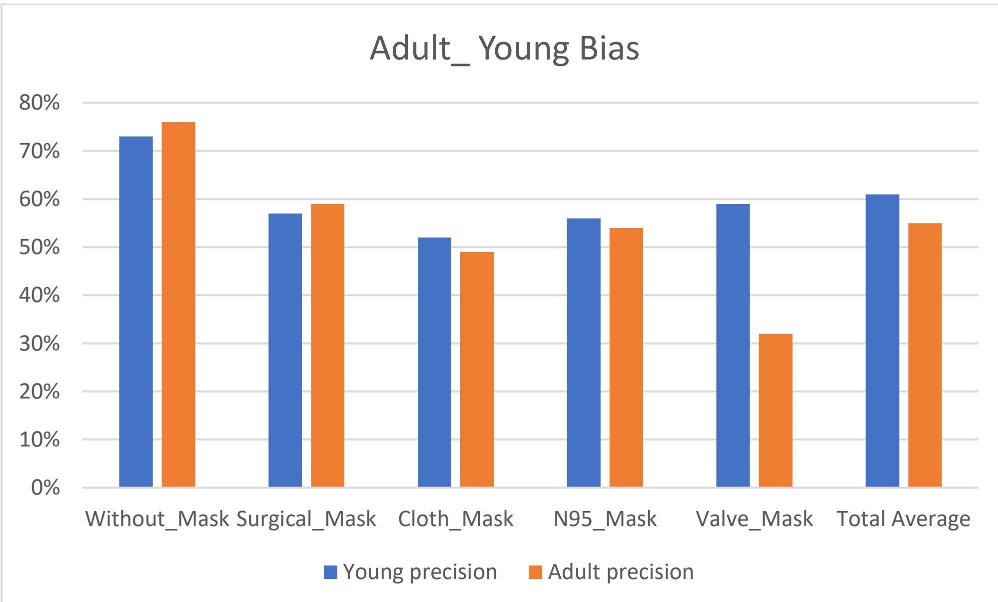


Figure : Gender bias test results and confusion matrix

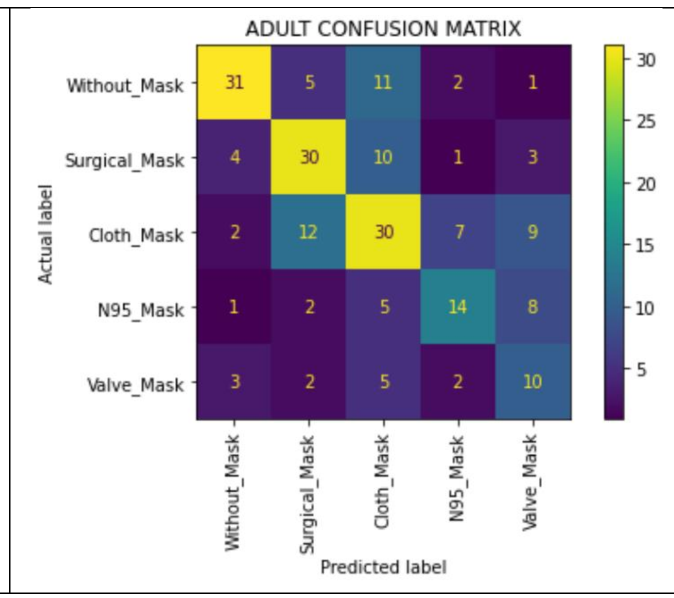
According to the below results and outcome, the bias has been enhanced. After improving the Bias, the trail run for the age bias in order to determine Bias of Adult and Young category.

As shown in below table and bar chart, there is 6% bias among the Adult and Young category. The considerable gab has been observed in valve mask category. Therefore, it is wise to increase the dataset in this category

	Young			Adult		
	precision	recall	f1-score	precision	recall	f1-score
Without_Mask	73%	77%	75%	76%	62%	68%
Surgical_Mask	57%	62%	59%	59%	62%	61%
Cloth_Mask	52%	45%	48%	49%	50%	50%
N95_Mask	56%	57%	57%	54%	47%	50%
Valve_Mask	59%	53%	56%	32%	45%	38%
Total Average	61%			55%		




```
              precision    recall  f1-score   support

 Without_Mask       0.76      0.62      0.68        50
Surgical_Mask       0.59      0.62      0.61        48
   Cloth_Mask       0.49      0.50      0.50        60
     N95_Mask       0.54      0.47      0.50        30
   Valve_Mask       0.32      0.45      0.38        22

     accuracy                           0.55       210
    macro avg       0.54      0.53      0.53       210
 weighted avg       0.57      0.55      0.55       210

[[31  5 11  2  1]
 [ 4 30 10  1  3]
 [ 2 12 30  7  9]
 [ 1  2  5 14  8]
 [ 3  2  5  2 10]]
```



```
              precision    recall  f1-score   support

 Without_Mask       0.73      0.77      0.75        75
Surgical_Mask       0.57      0.62      0.59        48
   Cloth_Mask       0.52      0.45      0.48        51
     N95_Mask       0.56      0.57      0.57        40
   Valve_Mask       0.59      0.53      0.56        30

     accuracy                           0.61       244
    macro avg       0.60      0.59      0.59       244
 weighted avg       0.61      0.61      0.61       244

[[58 10  5  1  1]
 [ 9 30  6  2  1]
 [ 4  7 23 12  5]
 [ 5  2  6 23  4]
 [ 3  4  4  3 16]]
```

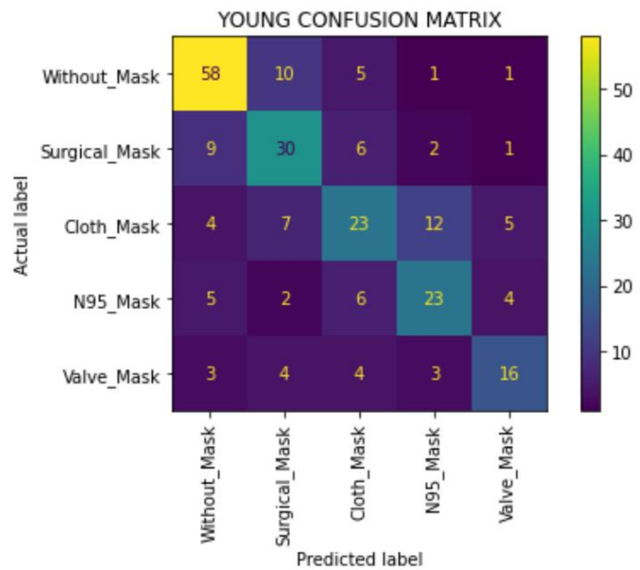
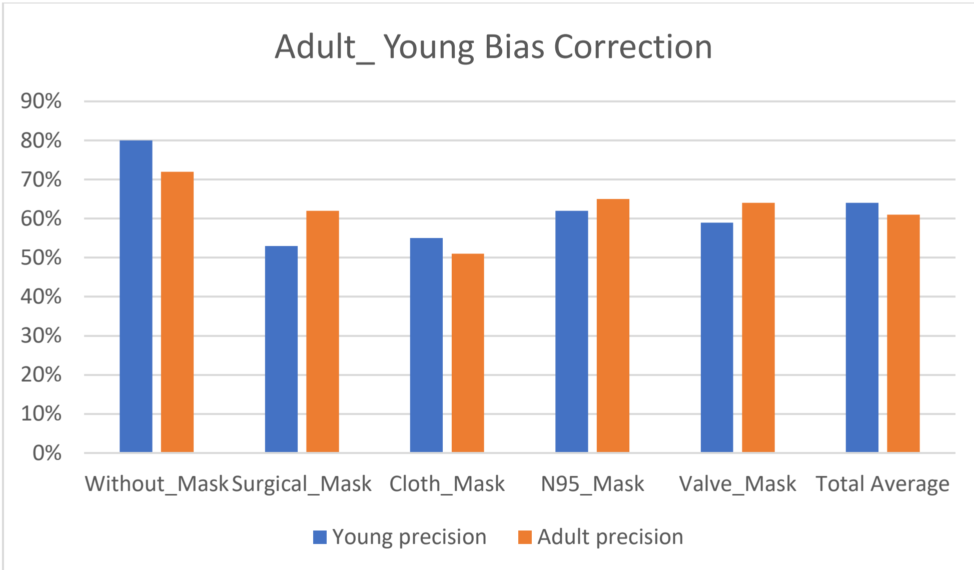


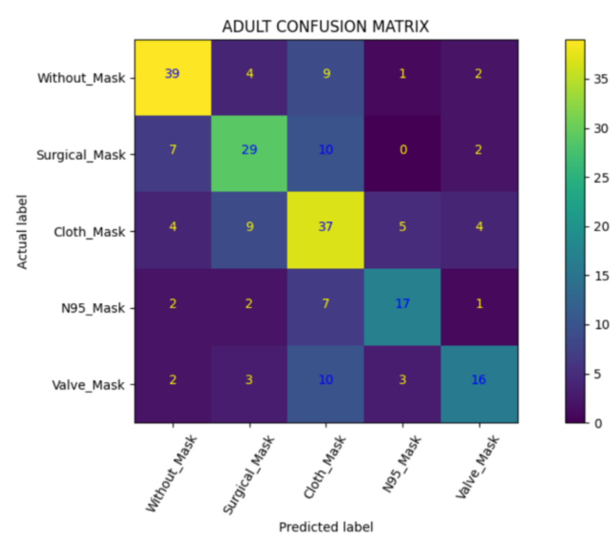
Figure : Adult and Young bias test results and confusion matrix

After increasing of dataset size by 148 images for the mention categories as well as doing dataset clean up for unclear images (18 images), following result has been achieved. It is well worth to mentioned that this is done in multiple steps to enhance the Bias as dataset size is not large enough to have better training model and stable result.

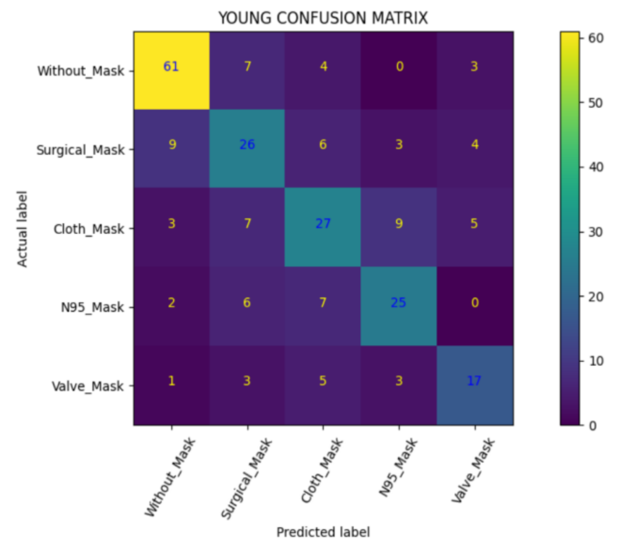
	Young			Adult		
	precision	recall	f1-score	precision	recall	f1-score
Without_Mask	80%	81%	81%	72%	71%	72%
Surgical_Mask	53%	54%	54%	62%	60%	61%
Cloth_Mask	55%	53%	54%	51%	63%	56%
N95_Mask	62%	62%	62%	65%	59%	62%
Valve_Mask	59%	59%	59%	64%	47%	54%
Total Average	64%			61%		




```
               precision    recall  f1-score   support

 Without_Mask       0.72      0.71      0.72        55
Surgical_Mask       0.62      0.60      0.61        48
   Cloth_Mask       0.51      0.63      0.56        59
     N95_Mask       0.65      0.59      0.62        29
   Valve_Mask       0.64      0.47      0.54        34

     accuracy                           0.61       225
    macro avg       0.63      0.60      0.61       225
 weighted avg       0.62      0.61      0.61       225

[[39  4  9  1  2]
 [ 7 29 10  0  2]
 [ 4  9 37  5  4]
 [ 2  2  7 17  1]
 [ 2  3 10  3 16]]
```



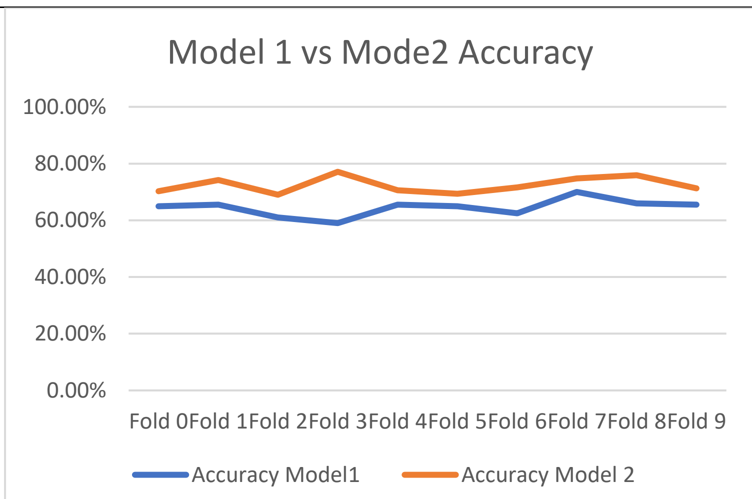
```
               precision    recall  f1-score   support

 Without_Mask       0.80      0.81      0.81        75
Surgical_Mask       0.53      0.54      0.54        48
   Cloth_Mask       0.55      0.53      0.54        51
     N95_Mask       0.62      0.62      0.62        40
   Valve_Mask       0.59      0.59      0.59        29

     accuracy                           0.64       243
    macro avg       0.62      0.62      0.62       243
 weighted avg       0.64      0.64      0.64       243

[[61  7  4  0  3]
 [ 9 26  6  3  4]
 [ 3  7 27  9  5]
 [ 2  6  7 25  0]
 [ 1  3  5  3 17]]
```

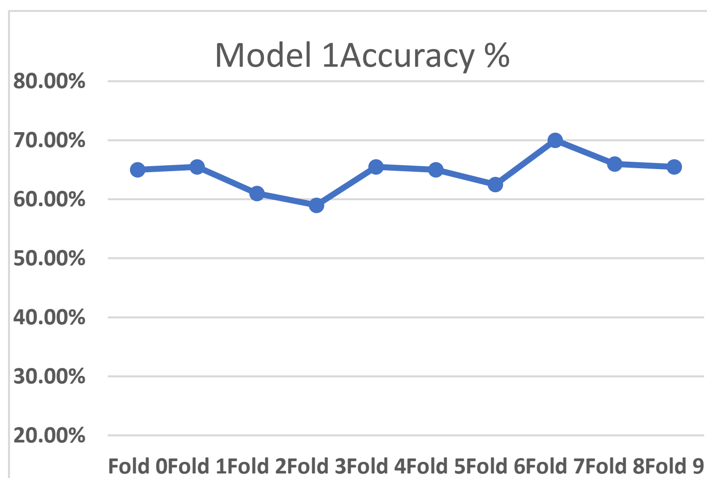


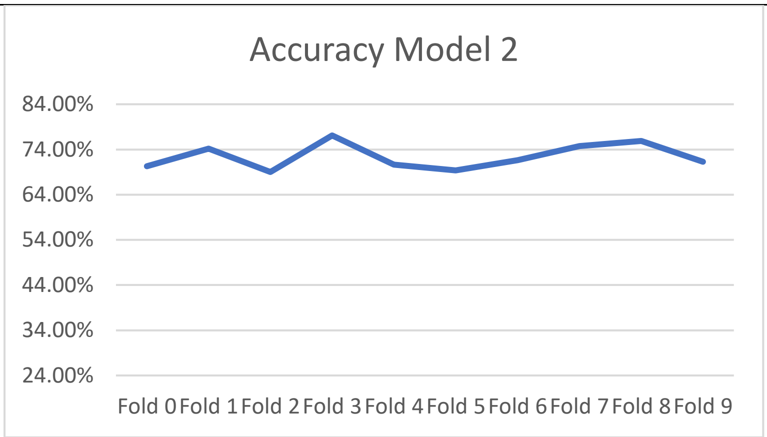
K Fold Cross Validation:

For evaluating and enhancing the performance our model, extra images are extracted from new resources as mentioned earlier and exposed the model to updated dataset. The model has been created based on pervious one and we run the K-fold cross validation. K-fold has been set to 10 in our model. In each K-fold phase, the training model has been saved and related test result was displayed, at the end of K-fold, we calculated mean average value of accuracy for K-fold.

Cross validation accuracy







Cross validation for K-Fold:

		1st Model			2nd Model		
Fold number	Category	precision	recall	f1-score	precision	recall	f1-score
0	Without_Mask	56%	74%	64%	78%	78%	78%
	Surgical_Mask	42%	31%	36%	68%	65%	67%
	Cloth_Mask	59%	35%	44%	91%	60%	72%
	N95_Mask	64%	9%	15%	73%	76%	75%
	Valve_Mask	33%	74%	46%	69%	92%	79%
1	Without_Mask	66%	72%	69%	85%	66%	74%
	Surgical_Mask	60%	52%	56%	58%	81%	68%
	Cloth_Mask	62%	57%	59%	83%	66%	73%
	N95_Mask	70%	48%	57%	66%	60%	63%
	Valve_Mask	68%	84%	75%	80%	81%	80%
2	Without_Mask	53%	44%	48%	69%	73%	71%
	Surgical_Mask	64%	56%	60%	64%	57%	60%
	Cloth_Mask	61%	63%	62%	84%	63%	72%
	N95_Mask	66%	68%	67%	55%	68%	61%
	Valve_Mask	74%	86%	79%	74%	83%	79%
3	Without_Mask	49%	69%	57%	77%	75%	76%
	Surgical_Mask	60%	36%	45%	76%	82%	79%
	Cloth_Mask	58%	65%	61%	83%	74%	78%
	N95_Mask	56%	76%	65%	52%	54%	53%
	Valve_Mask	77%	69%	73%	84%	83%	84%
4	Without_Mask	74%	56%	64%	63%	80%	71%
	Surgical_Mask	46%	54%	50%	62%	53%	57%
	Cloth_Mask	68%	65%	67%	70%	72%	71%
	N95_Mask	89%	77%	83%	72%	65%	68%
	Valve_Mask	70%	84%	76%	84%	82%	83%
5	Without_Mask	60%	57%	58%	73%	62%	67%

	Surgical_Mask	74%	48%	58%	64%	71%	68%
	Cloth_Mask	68%	64%	66%	77%	67%	72%
	N95_Mask	62%	86%	72%	60%	66%	63%
	Valve_Mask	65%	77%	71%	75%	78%	77%
6	Without_Mask	61%	50%	55%	76%	65%	70%
	Surgical_Mask	37%	50%	42%	53%	79%	64%
	Cloth_Mask	71%	76%	73%	73%	85%	79%
	N95_Mask	60%	67%	63%	89%	46%	60%
	Valve_Mask	83%	72%	78%	88%	76%	82%
7	Without_Mask	67%	41%	51%	73%	75%	74%
	Surgical_Mask	44%	58%	50%	65%	73%	69%
	Cloth_Mask	57%	52%	55%	84%	72%	78%
	N95_Mask	64%	75%	69%	70%	58%	63%
	Valve_Mask	77%	74%	76%	82%	84%	83%
8	Without_Mask	62%	58%	60%	76%	79%	78%
	Surgical_Mask	65%	65%	65%	67%	76%	71%
	Cloth_Mask	71%	90%	79%	83%	70%	76%
	N95_Mask	84%	66%	74%	71%	59%	65%
	Valve_Mask	73%	75%	74%	83%	86%	84%
9	Without_Mask	72%	59%	65%	79%	73%	76%
	Surgical_Mask	49%	43%	46%	62%	74%	67%
	Cloth_Mask	65%	83%	73%	78%	69%	74%
	N95_Mask	62%	60%	61%	83%	59%	69%
	Valve_Mask	75%	79%	77%	70%	74%	72%
10	Without_Mask	67%	68%	67%	76%	78%	76%
	Surgical_Mask	54%	48%	51%	68%	77%	72%
	Cloth_Mask	65%	71%	68%	82%	72%	73%
	N95_Mask	67%	65%	66%	72%	62%	67%
	Valve_Mask	74%	78%	76%	85%	84%	83%

Test result of 20% of dataset for the 1st model:

	precision	recall	f1-score	support
Without_Mask	0.72	0.78	0.75	145
Surgical_Mask	0.47	0.50	0.48	108
Cloth_Mask	0.62	0.56	0.59	93
N95_Mask	0.69	0.59	0.64	81
Valve_Mask	0.64	0.64	0.64	73
accuracy			0.63	500
macro avg	0.63	0.61	0.62	500
weighted avg	0.63	0.63	0.63	500

Test result of 20% of dataset for the 2^{nd} model:

	precision	recall	f1-score	support
Without_Mask	0.79	0.73	0.76	120
Surgical_Mask	0.61	0.73	0.67	148
Cloth_Mask	0.80	0.67	0.73	72
N95_Mask	0.84	0.60	0.70	68
Valve_Mask	0.67	0.74	0.70	114
accuracy			0.71	522
macro avg	0.74	0.69	0.71	522
weighted avg	0.72	0.71	0.71	522

```
[[ 88  18   1   2  11]
 [ 15 108   9   2  14]
 [  4  10  48   4   6]
 [  1  15   1  41  10]
 [  4  25   1   0  84]]
Accuracy of whole Model: 70 %
```

5- Conclusion and Future Prospects:

To sum up, following major actions were conducted to improve system performance.

1- Manipulating hidden parameters such as increasing the number of hidden layers, increasing the number of epochs, and decreasing the learning rate

2- The bias analysis was done for both gender and age, and related action has been taken to overcome the bias, however owing less number of images and dataset distribution within each category, it is not completely addressed.

3- K-Fold has been done for the phase 1 model and improved model, in order to have better estimation of trained model.

6- References:

Following are resource being referred and used for implementation as well as error debugging for the project:

- https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html
-
- https://stackoverflow.com/questions/50480689/pytorch-torchvision-brokenpipeerror-errno-32-broken-pipe

- https://towardsdatascience.com/how-i-built-a-face-mask-detector-for-covid-19-using-pytorch-lightning-67eb3752fd61

- https://discuss.pytorch.org/t/runtimeerror-input-type-torch-cuda-floattensor-and-weight-type-torch-floattensor-should-be-the-same/21782

- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

- https://pytorch.org/docs/stable/nn.html#convolution-layers

- https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html

- https://towardsdatascience.com/understanding-and-calculating-the-number-of-parameters-in-convolution-neural-networks-cnns-fc88790d530d

- https://stats.stackexchange.com/questions/296679/what-does-kernel-size-mean/339265

- https://www.analyticsvidhya.com/blog/2019/10/building-image-classification-models-cnn-pytorch/

- https://www.analyticsvidhya.com/blog/2020/10/what-is-the-convolutional-neural-network-architecture/

- https://towardsdatascience.com/pytorch-layer-dimensions-what-sizes-should-they-be-and-why-4265a41e01fd

- https://www.kaggle.com/datasets/sumansid/facemask-dataset?resource=download-directory

- https://github.com/AIZOOTech/FaceMaskDetection