COMP 6721 Project

AI Face Mask Detector - CNN

Final Report

TEAM NAME: AK_7

Name	ID	Specification
Akshay Dhabale	**40163636**	**Training Specialist**
Mrinal Rai	**40193024**	**Evaluation Specialist**
Siddhartha Jha	**40201472**	**Data Specialist**

PHASE 1

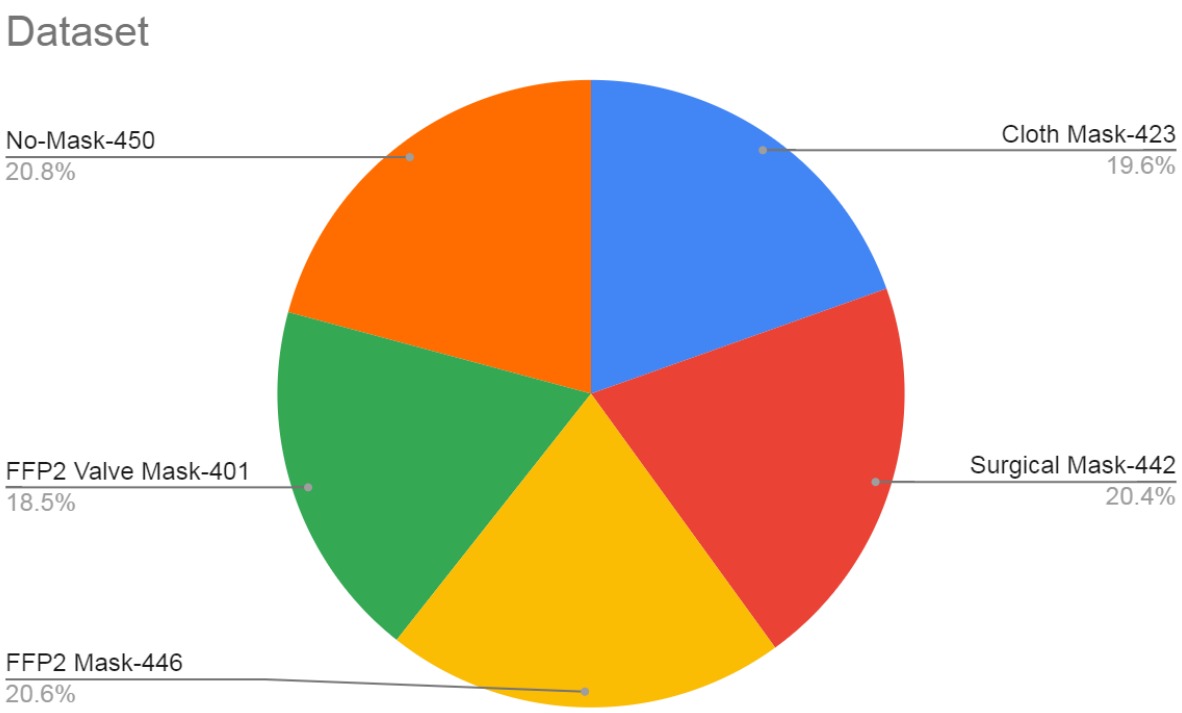
1. DATASET

1.1 COLLECTING DATA

Our dataset consists of 2162 images belonging to five classes. We train a CNN model to recognize five different classes (1) Person with a Cloth Mask, (2) Person with a Surgical-Mask, (3) Person with a FFP2-Mask, (4)Person with a FFP2-Valve-Mask and (5)Person without a Mask. The distribution of the dataset is provided in the pie chart below.

All the images of the dataset are stored in the “Dataset” directory and the format of all images is jpeg and png. The dataset includes five classes and each class contains approximately 400 images. Since the image resolution of each class is different, we used transforms. Resize function of torchvision to resize all the images to equal size of 128 x 128.

After the data preprocessing, we split the dataset randomly to train and test parts and use 75% of the data for training phase and the remaining 25% for testing phase. Source for the data is mentioned in the reference section.



1.2 PREPROCESSING

The following preprocessing steps were taken:

1)**Resize**: In order to feed the images to the CNN network, they were resized (128 px, 128 px).

2)**Horizontal-flips**: To increase randomness, images were flipped horizontally with a probability of 0.5

3)**Conversation to tensor**: Converts the images into an array of numbers, called torch tensor. Each pixel of the input RGB image is divided into three different pixels- red, blue, and green. This creates three different images. For each generated image, the pixel value is divided by 255 to range the pixel range from [0 255] to [0 1].

4)**Train-Test split**: The data set is split into 2 categories during runtime. The training dataset has 1622 images (75%) whereas the test dataset has 540 images (25%)

5)**Batch-loading**: Finally, images from both categories are loaded into a batch of size 32.

2. CNN Model Architecture

In this model, we have created a Convolution Neural Network which consists of 5 Convolution layers and 3 max-pooling layers and 3 fully connected layers. The model initiates processing with the tensor of shape [32,3,128,128]. This will produce the feature map with a depth of 96 at the first layer we have also used the kernel of size (5*5) and stride (2*2). We have used Relu as our activation function which will convert our negative values to 0. After this, we have the max-pooling layer which converts the input image into a set of rectangles and, for each such sub-region, outputs the maximum. This is also known as down-sampling. Thus, this model gives robustness and reduces the possibility of overfitting. Output from the first convolution block will be the input for the next convolution block. Then again, we have two more convolution layers with relu functions and at the last layer, we will again do the down-sampling using the pooling layer. Then the output of the convolution layer will be passed to the fully connected layer. For this, we have flattened the tensor to one dimension shape. All layers are classes in PyTorch's nn module, and model classes are inherited from nn. Module classes that handle the entire complexity of initializing model parameters (weighting), manipulating them, and storing them in memory. Below, the screenshot of the model summary is shown.

```
C:\Anaconda\envs\comp6721\python.exe "D:/Concordia Academics/Winter 2022/COMP 6721/COMP6721ProjectAK_07/app.py"
torch.Size([32, 3, 128, 128]) torch.Size([32])
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1          [-1, 96, 62, 62]           7,296
              ReLU-2          [-1, 96, 62, 62]               0
         MaxPool2d-3          [-1, 96, 31, 31]               0
            Conv2d-4         [-1, 256, 33, 33]         221,440
              ReLU-5         [-1, 256, 33, 33]               0
         MaxPool2d-6         [-1, 256, 16, 16]               0
            Conv2d-7         [-1, 384, 16, 16]         885,120
              ReLU-8         [-1, 384, 16, 16]               0
            Conv2d-9         [-1, 384, 16, 16]       1,327,488
             ReLU-10         [-1, 384, 16, 16]               0
           Conv2d-11         [-1, 256, 16, 16]         884,992
             ReLU-12         [-1, 256, 16, 16]               0
        MaxPool2d-13           [-1, 256, 7, 7]               0
           Linear-14                 [-1, 512]       6,423,040
             ReLU-15                 [-1, 512]               0
           Linear-16                 [-1, 256]         131,328
             ReLU-17                 [-1, 256]               0
           Linear-18                   [-1, 5]           1,285
================================================================
```

3. CNN Model Training

For training our CNN model, we are processing the data in batches for training we have specified the batch size of 32. This helps us to reduce the usage of memory to process large amounts of data. CNN model training has two phases, A forward phase, where the input is passed completely through the network. A backward phase, where gradients are backpropagated (backprop) and weights are updated.

At each epoch, using the forward function defined in the model class the input batch will go via all the layers specified in the function. After performing the forward pass, the loss is calculated using cross_entropy function of nn module for each prediction. The loss value or gradient is the error in the expected output and actual output. For this model, we have used Adam as an optimizer and a learning rate of 0.0001. This will adjust the weights in the model to minimize the loss. We have finally used the SoftMax activation function which will narrow down the output to probabilities between 0 and 1. In the last step of the model, all the five classes will get the probability value, and the highest among all is selected as the prediction by model.

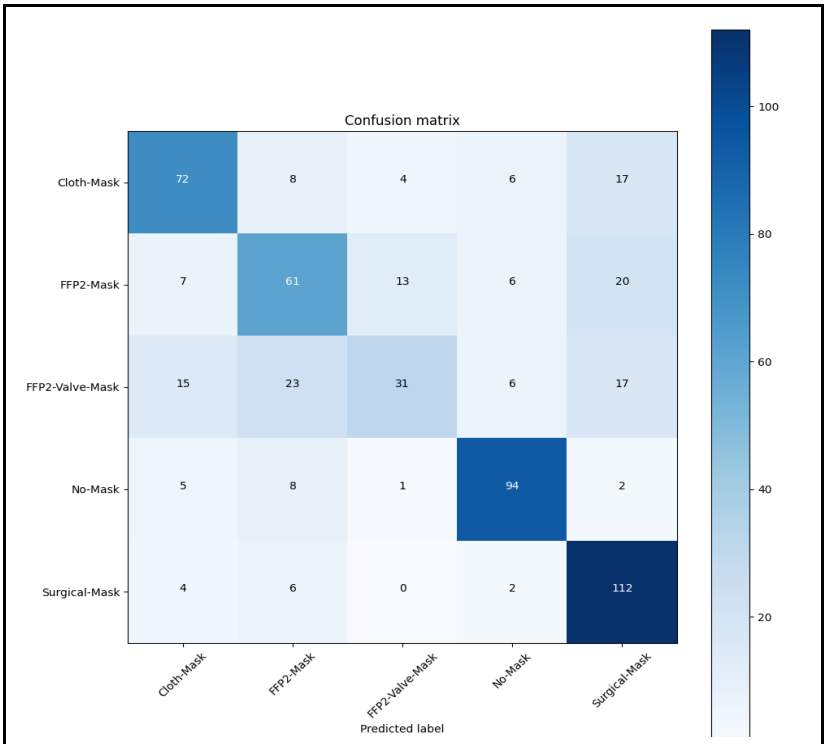
We trained the model for 15 epochs and results are as below:

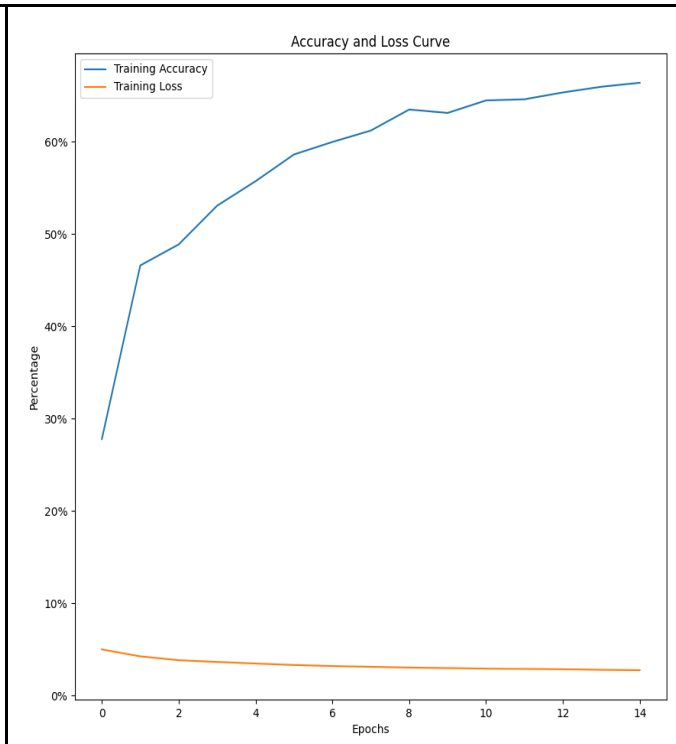
Epoch	Correct Predictions	Accuracy	Loss
1	451/1622	0.27805178791615287	0.05009848856896566
2	756/1622	0.46609124537607893	0.042621759392330295
3	793/1622	0.48890258939580766	0.038434573262011225
4	861/1622	0.530826140567201	0.03657967719432017
5	904/1622	0.5573366214549939	0.034847209654959736
6	951/1622	0.5863131935881628	0.03322386701215798
7	973/1622	0.5998766954377311	0.032124740017621646

8	993/1622	0.6122071516646116	0.031300552198537036
9	1030/1622	0.6350184956843403	0.030450540171010572
10	1024/1622	0.6313193588162762	0.02993686028679261
11	1046/1622	0.6448828606658447	0.029317023133819994
12	1048/1622	0.6461159062885327	0.02900136072009471
13	1060/1622	0.6535141800246609	0.028656306050708645
14	1070/1622	0.6596794081381011	0.028061387938724053
15	1077/1622	0.6639950678175093	0.02762886337963485

5. Evaluation

In the evaluation phase, we used our generated CNN model to analyze the test dataset of size 540 images from all the 5 categories of masks by plotting and interpreting a confusion matrix and calculating the accuracy, precision, recall and F1-measure. Below were the results obtained during testing phase:





Correct prediction: 370/540 and accuracy: 0.6851851851851852 and loss: 0.027018485135502285

	precision	recall	f1-score
Cloth-Mask	0.70	0.67	0.69
FFP2-Mask	0.58	0.57	0.57
FFP2-Valve-Mask	0.63	0.34	0.44
No-Mask	0.82	0.85	0.84
Surgical-Mask	0.67	0.90	0.77

Observations based on the results:

- Accuracy of 68.52% is achieved based on the CNN architecture and the dataset provided. As a future scope we will try to refine the model by adding more images doing hyperparameter tuning and experimenting with learning rate and number of epochs to increase the accuracy of the model
- FFP2 mask has the lowest precision and the confusion matrix indicates that the model confused FFP2 masks mostly with surgical and FFP2 valve masks. We would like to increase the quality of the dataset in the next phase by adding data augmentation techniques and removing ambiguous images from the data set
- No-Mask has the highest precision and the likely reasons are that it has the highest count of images in the dataset and it is easily recognisable by the model
- According to the classification report, Surgical masks have the highest recall which indicates that the model returns the most relevant results. On the other hand FFP2-valve masks had the lowest recall value as it detected fewer FFP2 valve masks.
- According to the Confusion matrix, the No-Mask category has the highest F1-score value which means the model at the current stage is best at predicting correct classes for the No-Mask category. On the other hand FFP2-valve masks have the lowest F1-score given its low precision and even lower recall value. We would try to increase the quality of the dataset as some images might not have enough pixels for the model to learn the features properly

PHASE 2

1. Improvement from Phase I

In the first Phase of the project we were able to achieve 68.51% of accuracy. So in order to improve our model, In phase 2 we tried different image augmentation techniques such as RandomVerticalFlip, RandomAdjustSharpness, RandomResizedCrop. But with our current dataset we could not get a better accuracy. In fact, it led to a decrease in accuracy by using the mentioned augmentation techniques so we reverted back to using only RandomHorizontal Flip. We also normalized our images with standard mean and deviation which we calculated from our current data. We also experimented with our CNN architecture by increasing the depth by adding the number of convolution layers. We tested this model both with the random training/test split and K-fold validation.

Finally we decided to change the learning rate from 0.00001 to 0.0001. Our model performed well with the 0.0001 learning rate and we got an accuracy of 71%. Below is the summarization of the changes done:

- Changed learning rate from 0.00001 to 0.0001
- Increased the depth of our Neural Network by adding 3 convolution layers
- Increased the number of Trainable parameters from 9,881,989 to 129,872,533

After making the above changes we trained the model using k fold validation for 10 splits. We achieved below results:

Pre-Bias Updated

K in 10-Fold CV	Accuracy %	Precision%	Recall%	F1-score%
1	69.36	70	68	69
2	73.41	71	68	69
3	69.94	67	67	67
4	65.89	66	65	64
5	71.67	72	70	70
6	72.67	75	73	72
7	72.67	70	71	69
8	76.74	75	77	76
9	75.58	74	75	73

10	69.76	68	70	69

Average Accuracy	71.1 %
Maximum Accuracy	76.74 %
Minimum Accuracy	65.89 %

Updated CNN Model Summary:

```
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1          [-1, 96, 62, 62]           7,296
              ReLU-2          [-1, 96, 62, 62]               0
         MaxPool2d-3          [-1, 96, 31, 31]               0
            Conv2d-4         [-1, 256, 33, 33]         221,440
              ReLU-5         [-1, 256, 33, 33]               0
         MaxPool2d-6         [-1, 256, 16, 16]               0
            Conv2d-7         [-1, 384, 16, 16]         885,120
              ReLU-8         [-1, 384, 16, 16]               0
            Conv2d-9         [-1, 984, 16, 16]       3,401,688
             ReLU-10         [-1, 984, 16, 16]               0
           Conv2d-11        [-1, 4048, 16, 16]      35,853,136
             ReLU-12        [-1, 4048, 16, 16]               0
           Conv2d-13        [-1, 2024, 16, 16]      73,740,392
             ReLU-14        [-1, 2024, 16, 16]               0
           Conv2d-15         [-1, 384, 16, 16]       6,995,328
             ReLU-16         [-1, 384, 16, 16]               0
           Conv2d-17         [-1, 384, 16, 16]       1,327,488
             ReLU-18         [-1, 384, 16, 16]               0
           Conv2d-19         [-1, 256, 16, 16]         884,992
             ReLU-20         [-1, 256, 16, 16]               0
        MaxPool2d-21           [-1, 256, 7, 7]               0
           Linear-22                 [-1, 512]       6,423,040
             ReLU-23                 [-1, 512]               0
           Linear-24                 [-1, 256]         131,328
             ReLU-25                 [-1, 256]               0
           Linear-26                   [-1, 5]           1,285
================================================================
Total params: 129,872,533
Trainable params: 129,872,533
Non-trainable params: 0
----------------------------------------------------------------
```

2. BIAS DETECTION AND ELIMINATION

In the second phase of this project assignment our goal was to look for any two of the following biases in our model: age, gender and race. We have studied the behavior of our improved model with the bias classes of age and gender. For this, we divided our dataset according to gender and age. For studying gender bias we divided our dataset into male and female categories. And for age, we divided our dataset into people of age between 0-40 and 41-100. For gender bias detection, we had 845 for female images and 878 images for the male. With this distribution we achieved the maximum accuracy of 75% for male dataset and for female dataset maximum accuracy of 73%. And average accuracy for male was 69% whereas in female dataset it was 64%. These were the results we noted from 10 fold validation. We can clearly see that the model is not performing well in the case of the female dataset.

So to increase the model performance we decided to add more data for the female dataset particularly in the no-mask and surgical classes. We also added 24 images in the FFP2 class for male dataset. So after data rebalancing we have 960 images in the female dataset and 923 images in male dataset. When we trained our model with the newly added images, we again used 10-fold validation on the entire male and female datasets. With the increase in the images our model performed quite well compared to the previous dataset. For male dataset we got the average accuracy of 70% and maximum accuracy was 80%. And for the female dataset we got the average accuracy of 70%, maximum accuracy of 76%. So with the increase in datasource our model performed well and had about the same accuracy as without bias datasets. In the below chart, we have all results from K-fold validation.

Male Pre-Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	75	74	72	72
2	65	60	61	58
3	73	66	68	66
4	69	66	66	66
5	66	65	64	64
6	65	61	63	61
7	70	69	71	68

Male Post-Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	70	65	66	66
2	66	65	62	63
3	67	66	64	64
4	75	74	72	72
5	80	71	72	70
6	66	63	63	62
7	71	72	67	67

8	74	72	70	71	**8**	74	72	70	71
9	57	50	55	51	**9**	72	71	72	70
10	75	74	68	69	**10**	68	64	72	65

Average Accuracy	69	Average Accuracy	70
Maximum Accuracy	75	Maximum Accuracy	80
Minimum Accuracy	57	Minimum Accuracy	66

Pre-Bias Male (Maximum Accuracy)

```
Correct prediction: 66/88
Accuracy: 0.75
Loss: 0.03262888504700227
                precision    recall  f1-score   support

     Cloth-Mask       0.77      0.71      0.74        14
      FFP2-Mask       0.75      0.40      0.52        15
FFP2-Valve-Mask       0.50      0.61      0.55        18
        No-Mask       0.93      0.96      0.95        27
  Surgical-Mask       0.76      0.93      0.84        14

       accuracy                           0.75        88
      macro avg       0.74      0.72      0.72        88
   weighted avg       0.76      0.75      0.74        88

[[10  0  3  0  1]
 [ 0  6  7  0  2]
 [ 2  2 11  2  1]
 [ 1  0  0 26  0]
 [ 0  0  1  0 13]]
```

Post-Bias Male (Maximum Accuracy)

```
Correct prediction: 72/90
Accuracy: 0.80
Loss: 0.02129837293095059
                precision    recall  f1-score   support

     Cloth-Mask       0.80      0.57      0.67        14
      FFP2-Mask       0.65      0.80      0.65        15
FFP2-Valve-Mask       0.65      0.70      0.65        19
        No-Mask       0.89      0.86      0.87        28
```

```
  Surgical-Mask       0.76      0.93      0.84        14

       accuracy                           0.80        90
      macro avg       0.71      0.72      0.70        90
   weighted avg       0.73      0.72      0.72        90

[[ 8  1  3  1  1]
 [ 0 12  1  0  2]
 [ 0  0 16  2  1]
 [ 2  0  2 24  0]
 [ 0  1  0  0 13]]
```

Female Pre-Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	65	61	64	60
2	56	54	58	52
3	73	64	65	64
4	65	67	60	60
5	56	56	45	47
6	60	51	51	49
7	70	60	66	61
8	65	64	60	58
9	71	59	61	59
10	62	56	56	54

Average Accuracy	64
Maximum Accuracy	73
Minimum Accuracy	56

Female Post-Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	65	63	62	62
2	76	74	76	73
3	71	66	68	66
4	70	61	63	60
5	68	62	59	60
6	62	59	61	58
7	73	66	70	68
8	74	65	65	64
9	71	71	66	65
10	72	69	65	64

Average Accuracy	70
Maximum Accuracy	76
Minimum Accuracy	62

Pre- Bias Female (Maximum Accuracy)

```
Correct prediction: 62/85, Accuracy: 0.7294117647058823
Loss: 0.025081693424898036
                precision    recall  f1-score   support

     Cloth-Mask       0.93      0.86      0.89        29
      FFP2-Mask       0.67      0.57      0.62        21
FFP2-Valve-Mask       0.14      0.14      0.14         7
        No-Mask       0.94      0.94      0.94        17
  Surgical-Mask       0.50      0.73      0.59        11
       accuracy                           0.73        85
      macro avg       0.64      0.65      0.64        85
   weighted avg       0.75      0.73      0.73        85
[[25  1  0  1  2]
 [ 1 12  5  0  3]
 [ 1  2  1  0  3]
 [ 0  0  1 16  0]
 [ 0  3  0  0  8]]
```

Post-Bias Female (Maximum Accuracy)

```
Correct prediction: 73/96
Accuracy: 0.7604166666666666
Loss: 0.023356092783312004
                precision    recall  f1-score   support

     Cloth-Mask       0.88      0.79      0.84        29
      FFP2-Mask       0.79      0.55      0.65        20
FFP2-Valve-Mask       0.47      0.80      0.59        10
        No-Mask       0.85      0.85      0.85        26
  Surgical-Mask       0.69      0.82      0.75        11
       accuracy                           0.76        96
      macro avg       0.74      0.76      0.73        96
   weighted avg       0.79      0.76      0.76        96

[[23  1  1  2  2]
 [ 0 11  7  1  1]
 [ 1  0  8  1  0]
 [ 2  1  0 22  1]
 [ 0  1  1  0  9]]
```

Similarly, for age bias detection, we had 1399 images for the people of age group 0-40 and 324 images for the people of age group 41-100. With this distribution we achieved the maximum accuracy of 76% for 0-40 dataset and maximum accuracy of 66% for 41-100 dataset. Average accuracy for 0-40 dataset was

69% whereas in 41-100 dataset it was 58%. These were the results noted from the 10 fold validation. We could distinctly see that the model was not performing well for the age group of 41-100 dataset. So to increase the performance of the model we decided to add more data for the age group of 41-100 dataset.

After cleaning the data for the people of age group 0-40 we now have 1190 images.We added more data for the age group of 41-100 which made upto 673 images. We re-trained our model with the newly added images and used 10-fold validation on the entire 0-40 and 41-100 age group datasets. With the increased number of images our model performed significantly well in comparison to the previous dataset. For the 0-40 dataset we got the average accuracy of 72% and maximum accuracy was 83%. On the other hand, for the 41-100 dataset we got the average accuracy of 65%, maximum accuracy of 76%. Hence, with the increase in datasource our model performed well and had nearly the same accuracy as the dataset without bias. In the below charts, we have all results from K-fold validation for the age bias.

Age Group 0-40 Pre Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	67	66	66	65
2	67	64	63	63
3	78	75	76	75
4	75	71	70	70
5	68	68	66	67
6	73	71	73	71
7	64	62	65	63
8	73	63	63	63
9	68	67	65	65
10	80	77	76	76

Average Accuracy	71

Age Group 0-40 Post-Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	70	64	68	64
2	71	59	60	59
3	69	67	68	66
4	68	68	66	66
5	71	69	68	67
6	77	77	74	75
7	69	70	69	70
8	72	69	73	70
9	67	67	64	64
10	83	81	78	79

Average Accuracy	72

Maximum Accuracy	77
Minimum Accuracy	64

Maximum Accuracy	83
Minimum Accuracy	67

Pre-Bias Age Group 0-40 (Maximum Accuracy)

```
Correct prediction: 111/139
Accuracy: 0.7985611510791367
Loss: 0.02239075560363934
                   precision    recall  f1-score   support

       Cloth Mask       0.76      0.90      0.83        29
        FFP2 Mask       0.70      0.80      0.74        20
FFP2 Mask - Valve       0.50      0.48      0.49        23
          No Mask       0.98      0.92      0.95        49
    Surgical Mask       0.93      0.72      0.81        18
         accuracy                           0.80       139
        macro avg       0.77      0.76      0.76       139
     weighted avg       0.81      0.80      0.80       139
[[26  0  2  0  1]
 [ 0 16  4  0  0]
 [ 4  7 11  1  0]
 [ 2  0  2 45  0]
 [ 2  0  3  0 13]]
```

Post-Bias Age Group 0-40 (Maximum Accuracy)

```
Correct prediction: 98/118
Accuracy: 0.8305084745762712
Loss: 0.018455430739006753
                 precision    recall  f1-score   support

     Cloth-Mask       0.73      0.80      0.76        20
      FFP2-Mask       0.81      0.88      0.85        34
FFP2-Valve-Mask       0.58      0.47      0.52        15
        No-Mask       0.94      0.97      0.96        35
  Surgical-Mask       1.00      0.79      0.88        14
       accuracy                           0.83       118
      macro avg       0.81      0.78      0.79       118
   weighted avg       0.83      0.83      0.83       118

[[16  2  2  0  0]
 [ 1 30  2  1  0]
 [ 2  5  7  1  0]
 [ 1  0  0 34  0]
 [ 2  0  1  0 11]]
```

Age Group 41-100 Pre Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	61	71	55	51
2	58	57	63	58
3	61	64	63	60
4	41	38	44	40
5	50	53	47	49
6	62	62	63	60
7	53	55	62	55
8	62	50	48	48
9	66	63	68	63
10	66	70	61	60

Average Accuracy	58
Maximum Accuracy	66
Minimum Accuracy	41

Age Group 41-100 Post-Bias

K in 10-Fold CV	Accuracy %	Precision %	Recall %	F1-score%
1	62	60	59	59
2	67	65	70	62
3	66	64	59	60
4	76	68	68	66
5	62	63	66	64
6	69	68	70	68
7	59	60	58	56
8	62	67	62	61
9	71	81	68	62
10	52	50	48	49

Average Accuracy	65
Maximum Accuracy	76
Minimum Accuracy	52

Pre-Bias Age Group 41-100 (Maximum Accuracy)

```
Correct prediction: 21/32
Accuracy: 0.65625
Loss: 0.025127174332737923
                  precision    recall  f1-score   support

       Cloth Mask       0.73      0.80      0.76        10
```

```
        FFP2 Mask       0.62      0.83      0.71         6
FFP2 Mask - Valve       1.00      0.25      0.40         4
          No Mask       0.62      0.83      0.71         6
    Surgical Mask       0.50      0.33      0.40         6
         accuracy                            0.66        32
        macro avg       0.70      0.61      0.60        32
     weighted avg       0.68      0.66      0.63        32

[[8 0 0 1 1]
 [0 5 0 0 1]
 [0 1 1 2 0]
 [1 0 0 5 0]
 [2 2 0 0 2]]
```

Post-Bias Age Group 41-100 (Maximum Accuracy)

```
Correct prediction: 44/58
Accuracy: 0.7586206896551724
Loss: 0.02746065012339888
                  precision    recall  f1-score   support

       Cloth Mask       0.86      0.80      0.83        15
        FFP2 Mask       0.83      0.50      0.62        10
FFP2 Mask - Valve       0.17      0.20      0.18         5
          No Mask       1.00      0.90      0.95        21
    Surgical Mask       0.54      1.00      0.70         7
         accuracy                           0.76        58
        macro avg       0.68      0.68      0.66        58
     weighted avg       0.81      0.76      0.77        58


[[12  1  2  0  0]
 [ 0  5  1  0  4]
 [ 2  0  1  0  2]
 [ 0  0  2 19  0]
 [ 0  0  0  0  7]]
```

3. K-FOLD CROSS-VALIDATION

Now that we eliminated bias from our dataset and improved our evaluation across the different classes using K-fold cross validation, we want to use 10-fold cross validation for our re-trained model.

K-fold cross-validation is a technique to evaluate a machine learning model on different data samples using a simple data resampling procedure. In this section we applied K fold on the model with 10 folds on the dataset, which splits into 9 trains and 1 test fold on the training data and trains the model accordingly. K-Fold Cross Validation ensures that on each fold a different set of Testing and Training data is provided to

the model. This ensures that the model is not overfitting to the same training data every time. K-Fold divides the dataset into pieces such that 1/k th data is used as testing. Based on this, the 10 fold applied model improves predictions compared to standard evaluation (fixed training/test split used in phase-1).

Then, we calculated the average accuracy from the accuracy of all folds.

K in 10-Fold CV	Accuracy %	Precision%	Recall%	F1-score%
1	75.93	75	74	73
2	70.58	72	71	71
3	79.67	78	79	78
4	73.65	71	72	71
5	68.88	71	65	66
6	77.95	77	78	77
7	74.19	74	76	74
8	72.58	75	72	72
9	71.50	73	71	71
10	81.18	79	80	80

Average Accuracy	74.61 %
Maximum Accuracy	81.18 %
Minimum Accuracy	68.88 %

Post-Bias Elimination (Maximum Accuracy)

```
Correct prediction: 151/186
Accuracy: 0.8118279569892473
Loss: 0.02300114240697635

                precision    recall  f1-score   support

     Cloth-Mask       0.82      0.84      0.83        38
      FFP2-Mask       0.83      0.76      0.80        46
FFP2-Valve-Mask       0.55      0.59      0.57        29
        No-Mask       0.96      0.92      0.94        51
  Surgical-Mask       0.80      0.91      0.85        22
```

```
              accuracy                           0.81       186
             macro avg       0.79      0.80      0.80       186
          weighted avg       0.82      0.81      0.81       186

[[32  0  3  1  2]
 [ 1 35  7  1  2]
 [ 6  5 17  0  1]
 [ 0  1  3 47  0]
 [ 0  1  1  0 20]]
```

As observed, we can state that our model is efficient to provide output with an average 74.61% accuracy for the given testing data. The average accuracy when using pre-bias data was around 71.1% but when using post-bias data after bias elimination the accuracy was around 74.61% with maximum accuracy of 81.18% and minimum accuracy of 68.88% which means the accuracy of the model is increased after retraining data and biases. We observed that the K-fold evaluation was efficient in testing the model as it provides consistent results with different training and testing datasets. After adding the post bias data we observe a slight increase in the K-fold results. This was mainly due to balancing the dataset and adding more testing images.

It can also be observed in the confusion matrix that the f1-score for all the classes of masks have been improved significantly. This was the result of adding images with better quality and removing skewed images as well as eliminating bias and increasing the depth of the CNN architecture.

	Pre-bias			**Post-bias fix**		
	0 - 40	**41 - 100**	**Total**	**0 - 40**	**41 - 100**	**Total**
Cloth	319	77	**396**	245	169	**414**
FFP2	279	70	**349**	242	123	**365**
FFP2-valve	235	65	**300**	205	149	**354**
No mask	398	51	**449**	375	110	**485**
Surgical	168	61	**229**	123	122	**245**
Total	**1399**	**324**	**1723**	**1190**	**673**	**1863**

	Pre-bias			Post-bias fix		
	Male	Female	Total	Male	Female	**Total**
Cloth	166	238	404	166	238	**404**
FFP2	121	228	349	145	228	**373**
FFP2-valve	194	104	298	194	104	**298**
No mask	268	181	449	268	263	**531**
Surgical	129	94	223	129	127	**256**
Total	**878**	**845**	**1723**	**923**	**960**	**1863**

References

[1] Cloth-Mask, Surgical-Mask images were collected from below sources:

shorturl.at/suNW4
shorturl.at/orzJR
shorturl.at/ovQTU
shorturl.at/hoAIK
shorturl.at/htwGW

[2] FFP2-Mask images were collected from Shutterstock.

shorturl.at/gsL38
shorturl.at/fnCJV
shorturl.at/foBFK
shorturl.at/ruABS

[3] No-Mask images were collected from 2 different Kaggle datasets

a.https://www.kaggle.com/vinaykudari/facemask by Vinay Kudari

b.https://www.kaggle.com/spandanpatnaik09/face-mask-detectormask-not-mask-incorrect-mask by Spandanpatnaik

[4] https://www.shutterstock.com/search/person+with+FFP2-Mask+with+valve

[5] https://www.kaggle.com/vinaykudari/facemask (Need to filter with valve)

[6] https://www.kaggle.com/datasets/sumansid/facemask-dataset

[7] https://www.kaggle.com/datasets/limyixen/facemask

[8] https://www.kaggle.com/datasets/prithwirajmitra/covid-face-mask-detection-dataset

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

[10] https://www.kaggle.com/datasets/maalialharbi/face-mask-dataset