Design, Implementation, and Characterization of a Proper Face Mask Usage Classification Model

Robin Jerome Reyes*, Geosef Viktor Uy*, Gabriel Nicolas Minamedez*, and Macario Cordel II

***Abstract*— The COVID-19 pandemic has prompted several interventions attempting to curb the viral spread, in which one of the most prominent measures is the wearing of face masks. While monitoring compliance to the protocol remains a challenge during the pandemic, several models have been developed to detect the presence of face masks in each person's face. However, a common limitation in present models is the lack of accounting for the proper usage of face masks. This study fills in those gaps by incorporating a multinomial face mask detection model to not only detect the presence of a face mask, but also the correctness in wearing it. The model is based on pretrained deep neural network architecture, fine-tuned using the Flickr Face HQ dataset and the MaskedFace-Net dataset containing 3000 images for each of the three categories: no mask, incorrectly worn mask, and correctly worn mask. The modelling involved preprocessing of images, face detection, mask detection, and classification, which achieved an accuracy of 99.11%, and a false discovery rate of 0.67%. This study can be significant in aiding the efforts of containing COVID-19, as determining the correctness of face mask usage can significantly improve surveillance in public spaces and prompt better contact tracing methods.**

I. Introduction

A report by the World Health Organization[1] showed the drastic effects of the Coronavirus disease (COVID-19) throughout the globe, with over 130 million cumulative cases since its existence, and around 3 million deaths. The situation has prompted a multitude of responses, both pharmaceutical [1] and non-pharmaceutical [2], to tackle the pandemic. Non-pharmaceutical interventions, in particular, pertain to the non-invasive nature of battling diseases by opting for products and protocols that are significantly helpful despite it being cost-effective [2]. One intervention in particular is the use of face masks, which has been proven to be effective in reducing the spread of COVID-19 [3], [4], when worn properly.

With the sheer amount of people that navigate various places, machine learning has made it possible to automate different tasks and monitor people in response to COVID-19. Face mask detection, which is achieved by incorporating face recognition to identify the presence of certain objects or features in an image [5], is very relevant in the current crisis to assist personnel in monitoring individuals that wear face masks when entering establishments or restricted areas.

All authors are from De La Salle University, 2401 Taft Avenue, Malate, Manila, Philippines

Corresponding authors: robin_jerome_reyes@dlsu.edu.ph, gabriel_nicolas_minamedez@dlsu.edu.ph, geosef_viktor_uy@dlsu.edu.ph, macario.cordel@dlsu.edu.ph

*All authors contributed equally to this research

[1]https://covid19.who.int/

However, present face mask detection endeavors [5]–[13] merely focused on the binary classification of whether or not the person is wearing face masks, which is limited in practice due to the fact that there may be multiple scenarios where face masks are incorrectly worn [14], and consequently, forfeiting the preventive effect that face masks hold against COVID-19. Though there has been exploration in the field of multinomial classification in face mask recognition [5], [7], these studies are focused mainly on the accuracy of classification rather than the misdetection of incorrectly worn mask as properly worn mask.

This work presents the design, implementation and characterization of a machine vision system that classifies whether the individual is wearing a face mask correctly, incorrectly, or nothing at all. Since the input involves images, a convolutional neural network (CNN) is utilized in the model, with the layers tinkered to determine the combination that provides classification prediction. Our proposed model is assessed using performance metrics i.e. accuracy and false discovery rate, confusion matrices, and saliency maps for fine-grain analysis.

The resulting model can provide substantial and practical application as it captures the true essence of face masks being effective when properly worn. It can be integrated in surveillance applications for public spaces to enforce the proper way of wearing a face mask so its benefits are maximized. Furthermore, it can be integrated in contact tracing efforts, determining scenarios where a person may have worn their face mask incorrectly and have an improved gauge on where the person was at highest risk of getting the virus.

This paper is organized as follows. Section I and II presents a background of present face mask detection and proper usage classification models. Section III explains the datasets used for training and validation. Section IV discusses the base model used for transfer learning. Section V presents the different design iterations made and the optimizations we conducted to come up with the final model design. Section VI focuses on discussing the results of the iterations done in the previous section. Finally, the summary, contribution, and recommendations are presented in the conclusion.

II. Related Works

A. Face Mask Detection

Face mask detection has unsurprisingly rose in interest because of the COVID-19 pandemic where mask-wearing is put at an emphasis. An abundance of face mask detection

models [5]–[10], [13] were developed to aid health, government, and other relevant authorities in the automation of determining the presence or lack thereof of a face mask on humans in a given image. Transfer learning is another method commonly employed in deep learning tasks to speed up progress on subsequent tasks by re-purposing and fine-tuning existing models. This is observed in RetinaFaceMask [10], a novel face mask detector that uses pre-trained ResNet and MobileNet for the backbone, complemented by pre-trained Imagenet [13] weights to resolve limited resources of data. Similarly, Loey et al. [6] proposed a hybrid deep learning model that uses ResNet50 for feature extraction and face detection, and consequently, decision trees and a support vector machine to classify the presence or absence of a face mask; they achieved 99.2% accuracy in [15] Bhandary's mask classifier dataset[2], and 100% in [16]. Militante & Dionisio introduced a real-time face mask recognition application [9] that made use of the VGGNet architecture and was able to achieve a 96% performance accuracy in detecting face masks.

B. Face Mask Usage Classification

Despite these advancements in face mask detection, all of the studies mentioned solely focused on the detection of face masks. As mentioned, this would be considered inadequate for actual, real-time scenarios wherein the correctness of the face mask wearing should be of high regard to limit the spread of the virus as much as possible. There have been attempts [5], [7], though limited in number, for this endeavor. Qin & Li [5] made use of an image super-resolution (SR) CNN for more fine-grained inputs, combined with the MobileNet-v2 for depthwise classification on three conditions: no face mask wearing (NFW), incorrect face mask wearing (IFW), and correct face mask wearing (CFW). Another study by Inamdar & Mehendale [7] made use of an R-CNN to classify face mask wearing on the same conditions.

The study by Qin & Li [5] centers around detecting faces using an SR network rather than purely classification accuracy. Our study focuses on the misclassification rate of the masks especially on the cases where the incorrect and no masks are classified as correctly worn face masks which is critical in detection systems, allowing people to violate face mask protocols undetected. Lastly, the dataset used by Qin & Li was imbalanced and had a large number of images for the correct mask class (3835) while having a scarce amount (134) for the incorrect mask class. Assessing the configurations of the aforementioned model, our study makes use of an innovative pre-trained model that has been fine-tuned to detect facial features from cropped images, since the model mainly predicts facial images coming from a balanced dataset. This is used as a base model via transfer learning, and the subsequent fully connected layers utilizes the weights to classify the correctness of face mask usage.

[2]https://github.com/prajnasb/observations

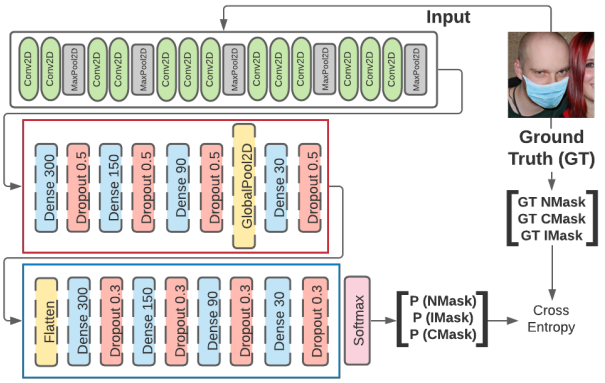

Fig. 1. Architecture of the proposed model. The layers with solid lines correspond to fixed parameters, which consists of the VGGFace base model, softmax layer, and cross entropy validation. The layers with broken lines are modified or removed in the design iterations. These consist of the dense layers, dropout layers, global pooling layers, and flatten layers.

III. Dataset

Our experiments use data sampled from two datasets. The first dataset is the Flickr Faces HQ (FFHQ) [17]. For our work, 3000 FFHQ images are randomly selected to account for the no mask samples for training, validation, and testing. The second dataset is the MaskedFace-Net [18], which is a collection of human faces with either correctly or incorrectly worn masks. Similar to the previous dataset, 3000 images are randomly selected for each class to account for the correctly and incorrectly worn mask samples. In total, 9000 images are used for this work, with each sample having different angles of faces.

The are three classes for this classification task, namely no mask, incorrectly worn mask, and correctly worn mask, hereunto called (*NMask*), (*IMask*), and (*CMask*), respectively, for brevity. A correctly worn mask case is defined as a face where the nose, mouth, and chin are all covered completely. Should at least one of those features be uncovered, it is then considered as an incorrectly worn mask case.

IV. Model Architecture

We use VGGFace [19] as the base architecture for this work, leveraging on its semantic rich feature extractor trained to detect face attributes. The VGGFace feature extractor is removed from its classifier, and is appended with dense layers to allow classification of the three classes, i.e. *NMask*, *IMask*, and *CMask*. In fine-tuning the model, the input data are scaled and then converted into TensorFlow Records, allowing for faster training. Fig. 1 illustrates this model.

In training the deep neural network (DNN) model, the layers of VGGFace are frozen to retain the weights of the model needed to extract the facial features. The softmax layer is placed at the end of the VGGFace model and the fully connected layers to generate the probability distributions of the predicted class which are used for classification. Lastly, categorical cross entropy is used as the loss function to

measure the difference between the ground truth and the predicted class.

In evaluating the initial DNN model, we use accuracy as one of the performance metrics for the classifier. The balanced number of images for each class from the dataset justifies the use of this metric since it does not take into account the class distribution. Additionally, the tests are conducted through k-folds cross validation in order to ensure that each data in the dataset can be evaluated and to ensure that our model does not only perform well in one set of training data. For the final model, we use the metrics of accuracy and false discovery rate, as well as confusion matrices to measure the final model performance.

V. Experiments

A. First Stage Classifier Optimization

The first set of experiments examines the appropriate number of dense layers, global pooling layers, and dropout layers. Each dense layer connects all neurons of the previous layer into the next layer, and takes into account the weights to be trained in these corresponding layers. As an example, a $7 \times 7 \times 512$ running through a dense layer of 300 nodes would take into account all 512 of the 7×7 instances per node, producing a $7 \times 7 \times 300$ image. Global pooling layers are also added to obtain each channel's definite amount out of all the pixels of the image. A $7 \times 7 \times 300$ implementing a global average pooling layer would obtain the average of each 7×7 feature possible, and would run across all 300 channels, resulting in 300 values. Finally, a dropout layer is used for regularization, which disregards random nodes depending on the rate specified. This section presents the experiments that lead to the first stage classifier in the red box portion of Fig. 1.

For the hyperparameters used in training the feature extractor and the first stage classifier, the Adam optimizer is utilized, having an initial learning rate of $1e{-}3$, exponential decay rate for the first and second moment estimates being β_1 = 0.900 and β_2 = 0.999, respectively, with number of epochs = 3. The number of layers L and nodes N for each dense layer are varied, evaluated and compared in terms of accuracy. We iterated using one to four dense layers with the number of nodes equal to 30, 90, 150 and 300, respectively. Please refer to Table I for the summary of the classification performance.

The experiments showed that using four dense layers provides the best accuracy. For the number of nodes used, the results found consistent training and validation accuracy, with the best number of nodes per layer being 300, 150, 90 and 30, for the l_1, l_2, l_3 and l_4, respectively.

To further tune the model, we check the model performance when global average pooling and global max pooling are used. Results show that adding the latter will significantly increase the performance from 93.73% to 95.40% accuracy. We also investigated the effect of adding a max pooling layer after l_1, l_2, l_3 and l_4. Results also suggest that a max pooling layer after l_3 will further improve the model performance. Using dropout = 0.1 also reveals better performance.

TABLE I

First Round of Model Experiments and Accuracy. The performance in bold is the best per experiment/iteration. L is the number of dense layers and N is the number of nodes. Layers are denoted as l_n where n is the nth layer.

Parameters α = $1e{-}3$ Epochs = 3	Mean Accuracy (Train)	Mean Accuracy (Validation)
Number of dense layers iterations		
$L = 1$, $N = 30$	92.38%	93.21%
$L = 2$, $N = 90, 30$	94.26%	92.39%
$L = 3$, $N = 150, 90, 30$	93.53%	92.72%
$L = 4$, $N = 300, 150, 90, 30$	93.74%	**93.75%**
Number of nodes iterations, $L = 4$		
$N = 30, 15, 9, 3$	93.12%	93.12%
$N = 50, 25, 15, 5$	94.17%	93.11%
$N = 100, 50, 30, 10$	93.39%	93.21%
$N = 150, 75, 45, 15$	93.94%	93.32%
$N = 200, 100, 60, 20$	94.21%	91.15%
$N = 300, 150, 90, 30$	93.74%	**93.75%**
Pooling iterations		
GlobalPool2D = Ave (see Fig. 1)	94.29%	93.73%
GlobalPool2D = Max (see Fig. 1)	97.16%	**96.40%**
Max pooling layer after l_3	98.62%	**97.40%**
Dropout iterations		
Dropout 0.1	99.37%	**98.35%**
Dropout 0.3	98.93%	97.90%
Dropout 0.5	98.84%	98.26%

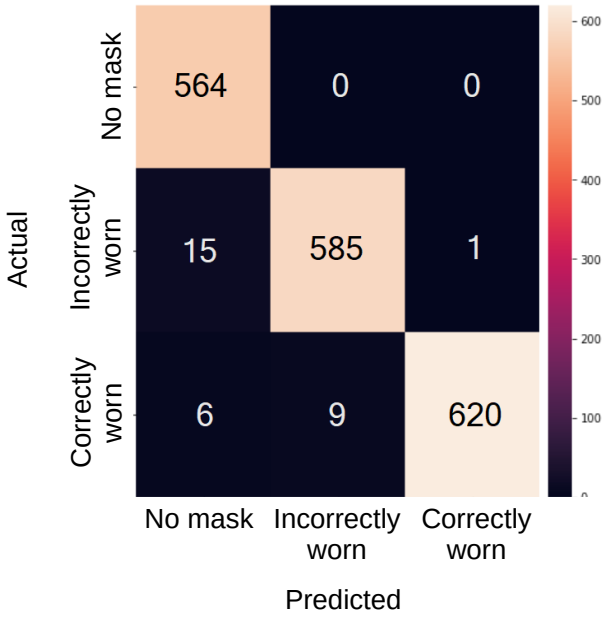

Fig. 2. Confusion matrix generated from the initial model with Stage 1 classifier, with 98.28% validation accuracy. Note that there are 31 incorrect classifications.

In summary for the first stage 4-layer classifier, we chose a global max layer after l_3, and a dropout layer with a rate of 0.1 after l_4 (see Table 1).

To evaluate the performance of our model, confusion matrices are checked based on the predictions to determine

TABLE II

SECOND ROUND OF MODEL EXPERIMENTS AND ACCURACY. THE PERFORMANCE IN BOLD IS THE BEST PER EXPERIMENT/ITERATION. LAYERS ARE DENOTED AS l_n WHERE n IS THE nTH DENSE LAYER.

Parameters α = $1e{-}3$ Epochs = 10	Mean Accuracy (Train)	Mean Accuracy (Validation)
Stage 1 & 2 Dropout iterations with Flatten		
Stage 1 + Dropout 0.5, Flatten, and Stage 2 + Dropout 0.5	94.74%	98.39%
Stage 1 + Dropout 0.5, Flatten, and Stage 2 + Dropout 0.3	96.28%	**98.56%**
Stage 1 + Dropout 0.5, Flatten, and Stage 2 + Dropout 0.1	96.75%	98.25%
Stage 1 + Dropout 0.3, Flatten, and Stage 2 + Dropout 0.3	97.57%	98.25%
Stage 1 + Dropout 0.3, Flatten, and Stage 2 + Dropout 0.1	98.17%	96.19%
Stage 1 + Dropout 0.1, Flatten, and Stage 2 + Dropout 0.1	98.88%	98.18%
Flatten after Base model, number of epochs		
epochs = 10	96.32%	98.13%
epochs = 15	97.74%	98.21%
epochs = 20	98.37%	**98.39%**
epochs = 25	98.75%	98.24%
α = $1e{-}4$, Epochs iterations		
epochs = 10	86.74%	95.49%
epochs = 15	91.01%	96.69%
epochs = 20	92.67%	97.39%
epochs = 25	94.75%	**97.93%**

the critical categories where the model tends to misclassify, as shown in Fig. 2. Despite the high accuracy reported in Table I, the confusion matrix reveals that the model accurately classified *NMask* but obtained high numbers of misclassification in both *CMask* and *IMask* classes.

B. Second Stage Classifier Optimization

The second set of experiments are conducted, where the configurations of Stage 1 are still in place, but instead of applying a global max pooling layer, the output is flattened from $7 \times 7 \times 30$ to 1470 values in a one-dimensional array. This prompted a duplicate of Stage 1 dense layers to allow more complexity of prediction after flattening, in which these layers are used for Stage 2 optimization. Fig. 1 represents the second stage classifier through the blue box portion. Additionally, dropout layers are added after each dense layer to serve as regularization [20], finalizing on a dropout rate of 0.5 for the Stage 1 dense layers, and 0.3 dropout rate for Stage 2. From this point on, we shall refer to Stage 1 with regards to its dense and dropout layers only, excluding its pooling layers.

The flatten layer is also subjected to experiments, moving it to the beginning of the fully connected layer, compared to the previous experiment where the flatten layer is between Stage 1 and Stage 2. An increase in trainable parameters led to a decrease in accuracy, prompting us to retain the original configuration. Results also generated unfavorable outcomes for the adjustment of learning rate and number of epochs to

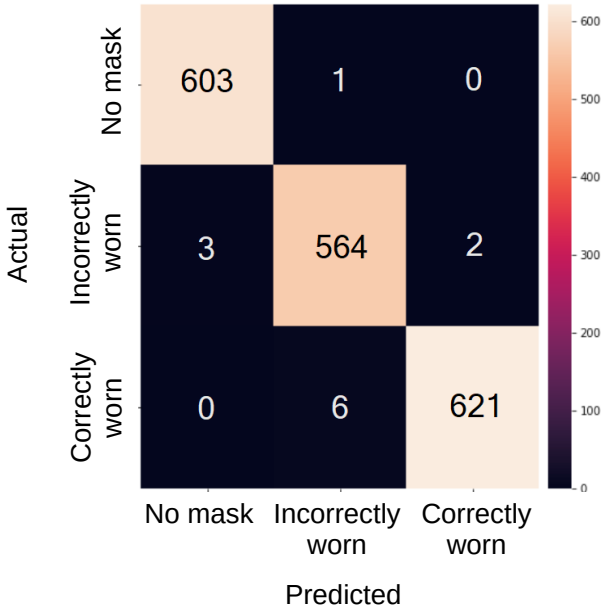

Fig. 3. Confusion matrix generated from the final model with Stage 1 and Stage 2 classifiers, obtaining a 99.33% validation accuracy. Note that there is a significant decrease in incorrect classifications from 31 to 12.

$1e{-}4$ and 15, 20, and 25 respectively. We decided to revert to the initial parameters of $1e{-}3$ learning rate and 10 epochs.

In summary, the final model uses a combination of Stage 1 with dropout rates of 0.5 for each layer, a flatten layer, and Stage 2 with dropout rates of 0.3 for each layer. The learning rate is $1e{-}3$ that runs through 10 epochs. This configuration is chosen through obtaining the highest validation accuracy through 3 folds.

Images with failed predictions are then extracted and analyzed to determine their nature. Furthermore, they are inputted and converted into saliency maps to establish a ranking behind the pixels based on their contribution to the final classification from the model.

VI. DISCUSSION

The confusion matrix shown in Fig. 3 shows the details of the classification performance of the designed model. Among the misclassifications, predicting *NMask* as *CMask*, and *IMask* as *CMask* were given emphasis since these errors can pose significant risks of viral spread. The matrix shows that no *NMask* samples were misclassified as *CMask*, and there are only 0.35% of the *IMask* samples were misclassified as *CMask*.

Saliency maps were also used to visualize the decision-making process of the model by showing which image region has the higher importance or weight. We superimposed the saliency map over the input image, as heat map, such that pixels closer to red have higher relevance to the classification process, whereas pixels to blue factor means lower relevance.

As visualized in Fig. 4, generally, regardless of the input class, the image regions with the highest importance are those in the nose and mouth parts, represented by a red color

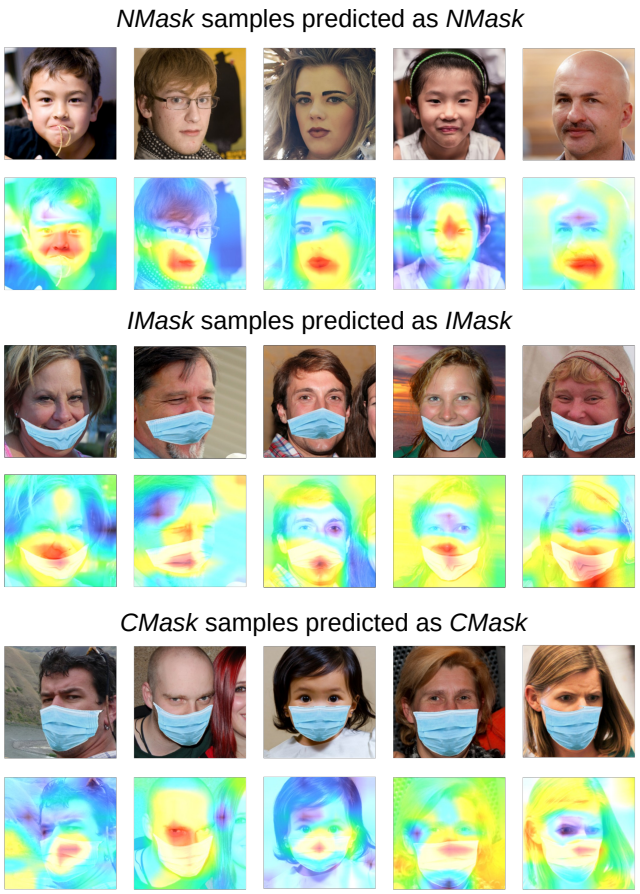

Fig. 4. Correct predictions and their corresponding saliency maps. Five images from the validation set for each class (*CMask,IMask, NMask*) were selected.

Fig. 5. Incorrect Mask Predictions. The images were extracted based on the validation dataset, which had 2 predictions of *IMask* as *CMask*. The two samples are *IMask* cases because the chins are not properly covered by the face mask.

intensity. These regions are the critical areas for determining the proper usage of face masks since these regions are exposed when there is no face mask, or covered either fully or partially by a face mask.

In Fig. 4, we show that only the mouth and the nose regions are important in accurate classifications. In real-life scenarios, however, it is more practical to focus on riskier cases where a person incorrectly wearing a face mask is misclassified as correctly wearing it. The two misclassified samples in Fig. 5 show quite challenging cases where *IMask* are classified as *CMask*; they are supposedly *IMask* since these samples cover only a small portion of the chin, contradicting the original definition of a *CMask* usage. Because of this, they leave a smaller differentiating area for proper face mask usage classification. To increase the coverage of the salient parts that can improve classification performance for such cases, more *IMask* samples with chins not properly covered are needed during training.

Fig. 6. Face mask detection model tested against real life instances of face mask wearing. Texts in red are misclassifications.

To showcase the capabilities of the model and benchmark in actual scenarios, different face mask images were used to incorporate not only surgical masks, but also N95 and cloth masks of varying colors. *NMask* and *CMask* categories were taken from a public dataset[3] of different face masks, and actual photos were taken by the authors to identify *IMask* images since there is a scarcity in images that show improper usage of non-surgical face masks.

It is shown in Fig. 6 that the model was able to achieve perfect accuracy in terms of both *NMask* and *IMask*, making sure that non-compliant individuals are detected. However, a possible improvement would be to focus on the prediction of the *CMask* category, since 2 out of 5 actual face mask images were wrongly classified as *NMask* and *IMask* respectively. Nevertheless, the model can predict proper usage of face masks despite being trained solely on synthetic face masks.

After obtaining the saliency maps, the test dataset was predicted and the model obtained a final accuracy of 99.11%, which is similarly high along with the validation accuracies obtained. To further expand on its evaluation, a false discovery rate (FDR) focused for the *CMask* images is also indicated to provide insight on how robust the model is with detecting non-compliant individuals. In the case of *CMask* predictions, the FDR is defined in Eq. (1) as

[3]https://www.kaggle.com/dhruvmak/face-mask-detection

$$FDR = \frac{FP}{FP + TP} \quad (1)$$

wherein all of the misclassified images predicted as *CMask* interpreted as FP or false positives are divided by the total *CMask* predictions generated, which is the sum of FP and the TP or true positives. Table III illustrates the categories considered, as well as the test accuracy across relevant studies [5], [7], [8] referenced in the related works.

TABLE III
COMPARISON TO RELATED WORKS. PERFORMANCE IN BOLD ARE THE BEST FOR THAT COLUMN

Model	*CMask*	*NMask*	*IMask*	Performance	FDR
[8]	✓	✓		99.19%	**0.65%**
[5]	✓	✓	✓	98.70%	0.83%
[7]	✓	✓	✓	98.60%	N/A
Ours	✓	✓	✓	99.11%	**0.67%**

When compared to other works that perform face mask detection, the model appears to achieve comparable performance, if not higher accuracy. Loey and colleagues [8] were able to achieve the highest accuracy for binomial classifications. In comparison to the few studies [5], [7] that tackle the correctness of face mask usage, the proposed model was able to achieve the highest accuracy at 99.11%. Furthermore, we also looked into the FDR of the different models, which gave insight on possible violation of health protocols; this was a metric not taken into account particularly by the other multinomial studies. Our study achieved a competitive FDR which is equal to 0.67%.

VII. CONCLUSION

Multinomial classification of face mask usage based on DNN was empirically designed, implemented, evaluated in terms of classification accuracy, and visualized through saliency maps for model interpretation. Rather than face mask detection alone, we presented a model trained to classify input images into *NMask*, *IMask*, or *CMask*. Unlike others, our work is more focused on optimizing false discovery rate (FDR), the metric that measures the misclassification rate of incorrectly worn masks and no face mask cases as correctly worn masks. Our model obtained high accuracy and low FDR for the test dataset. Also, preliminary tests on face mask usage in the wild showed no misclassification of 10 *NMask* and *IMask* cases as *CMask*. This is especially important in enforcing COVID-19 health protocols for proper usage of face masks.

The following are for future work. First, the number of folds in the cross validation should be increased for a clearer picture of the model performance in different training samples. Second, for a more robust model, the addition of other variants of masks such as cloth masks and N95 masks, taken from the wild, could adapt and improve the model performance. However, collecting incorrectly worn face masks might be a challenge. Finally, for the system deployment, an end-to-end system will be implemented which considers multiple faces in a scene.

REFERENCES

[1] S. P. Kaur and V. Gupta, "Covid-19 vaccine: A comprehensive status report," *Virus Research*, vol. 288, p. 198114, 2020.

[2] N. Askitas, K. Tatsiramos, and B. Verheyden, "Estimating worldwide effects of non-pharmaceutical interventions on COVID-19 incidence and population mobility patterns using a multiple-event study," *Scientific Reports*, vol. 11, no. 1, p. 1972, Dec 2021.

[3] H. J. Schünemann, E. A. Akl, R. Chou, D. K. Chu, M. Loeb, T. Lotfi, R. A. Mustafa, I. Neumann, L. Saxinger, S. Sultan, and D. Mertz, "Use of facemasks during the covid-19 pandemic," *The Lancet Respiratory Medicine*, vol. 8, pp. 954–955, 2020.

[4] N. H. Leung, D. K. Chu, E. Y. Shiu, K. H. Chan, J. J. McDevitt, B. J. Hau, H. L. Yen, Y. Li, D. K. Ip, J. S. Peiris, W. H. Seto, G. M. Leung, D. K. Milton, and B. J. Cowling, "Respiratory virus shedding in exhaled breath and efficacy of face masks," *Nature Medicine*, vol. 26, pp. 676–680, 2020.

[5] B. Qin and D. Li, "Identifying facemask-wearing condition using image super-resolution with classification network to prevent covid-19," *Sensors*, vol. 20, Sept 2020.

[6] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "Fighting against covid-19: A novel deep learning model based on yolo-v2 with resnet-50 for medical face mask detection," *Sustainable Cities and Society*, vol. 65, 2021.

[7] M. Inamdar and N. Mehendale, "Real-time face mask identification using facemasknet deep learning network," *SSRN Electronic Journal*, 2020.

[8] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the covid-19 pandemic," *Measurement*, vol. 167, Jan 2021.

[9] S. V. Militante and N. V. Dionisio, "Real-time facemask recognition with alarm system using deep learning," in *2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC)*. IEEE, Aug 2020.

[10] M. Jiang, X. Fan, and H. Yan, "Retinamask: A face mask detector," *Computer Vision and Pattern Recognition*, 2020.

[11] G. J. Chowdary, N. S. Punn, S. K. Sonbhadra, and S. Agarwal, "Face mask detection using transfer learning of inceptionv3," *Computer Vision and Pattern Recognition*, 2020.

[12] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, and J. Hemanth, "Ssdmnv2: A real time dnn-based face mask detection system using single shot multibox detector and mobilenetv2," *Sustainable Cities and Society*, vol. 66, Mar 2021.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2009.

[14] M. Machida, I. Nakamura, R. Saito, T. Nakaya, T. Hanibuchi, T. Takamiya, Y. Odagiri, N. Fukushima, H. Kikuchi, S. Amagasa, T. Kojima, H. Watanabe, and S. Inoue, "Incorrect use of face masks during the current covid-19 pandemic among the general public in japan," *International Journal of Environmental Research and Public Health*, vol. 17, pp. 1–11, 2020.

[15] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei, H. Chen, Y. Miao, Z. Huang, and J. Liang, "Masked face recognition dataset and application," 2020.

[16] E. Learned-Miller, G. B. Huang, A. RoyChowdhury, H. Li, and G. Hua, "Labeled faces in the wild: A survey," in *Advances in Face Detection and Facial Image Analysis*. Springer International Publishing, Jan 2016, pp. 189–248.

[17] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," 2019.

[18] A. Cabani, K. Hammoudi, H. Benhabiles, and M. Melkemi, "Maskedface-net – a dataset of correctly/incorrectly masked face images in the context of covid-19," *Smart Health*, vol. 19, Mar 2021.

[19] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *British Machine Vision Conference*, 2015.

[20] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," 2012.