Face Mask Detection Using CNN

Joydep Dhar, Chowdhury, Md. Rifat, Munshi Azmain Anan, Zubaida Mst Khatun,

1. Project Objective:

Our research proposes a condensed method for facemask recognition and customized notice for people who are not wearing one. To do this, we used Kaggle datasets for the suggested system/model's training and evaluation. The device can tell if someone is wearing a facemask or not since it works in real time. The technology gives the person a notice message directly if they are not donning a facemask. The facemask is taken out of real-time, public faces and sent into a convolutional neural network (CNN) for analysis according to the suggested approach. Overall, the suggested system offers a trustworthy and effective method for facemask detection, which is especially pertinent given the current state of world health.

2. Project Methodology:

In order to predict whether a person has put on a mask, the model requires learning from a well- curated dataset, as discussed later in this section. The model uses Convolution Neural Network layers (CNN) as its backbone architecture to create different layers. Along with this, libraries such as OpenCV, CoLab, Jupyter Notebook and Streamlit are also used. The proposed model is designed in three phases: Data pre-processing, CNN model training and Applying face mask detector as described in Fig.1.

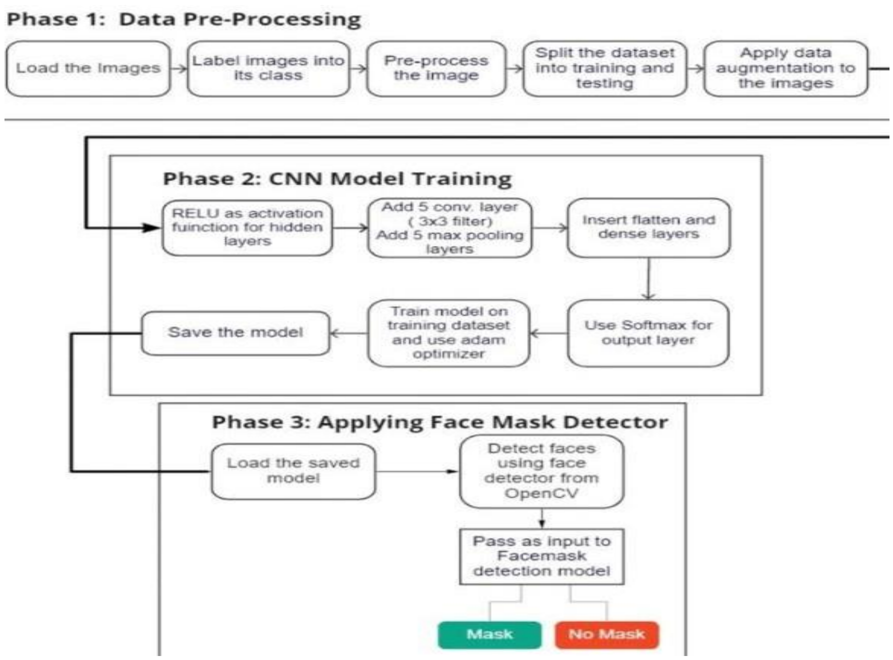


Fig. 1: Proposal Architecture

2.1. Data Collection Procedure:

The dataset used in this research was collected in various picture formats such as JPEG, PNG, and others. Figure 1 exhibits the sample of the dataset. It was a mixture of different open-source datasets and images, including Kaggle's dataset for Face Mask Detection. As a result, there were different varieties of images with variations in size and resolution. All the photos were from open-source resources, out of which some resemble real-world scenarios, and others were artificially created to put a mask on the face.

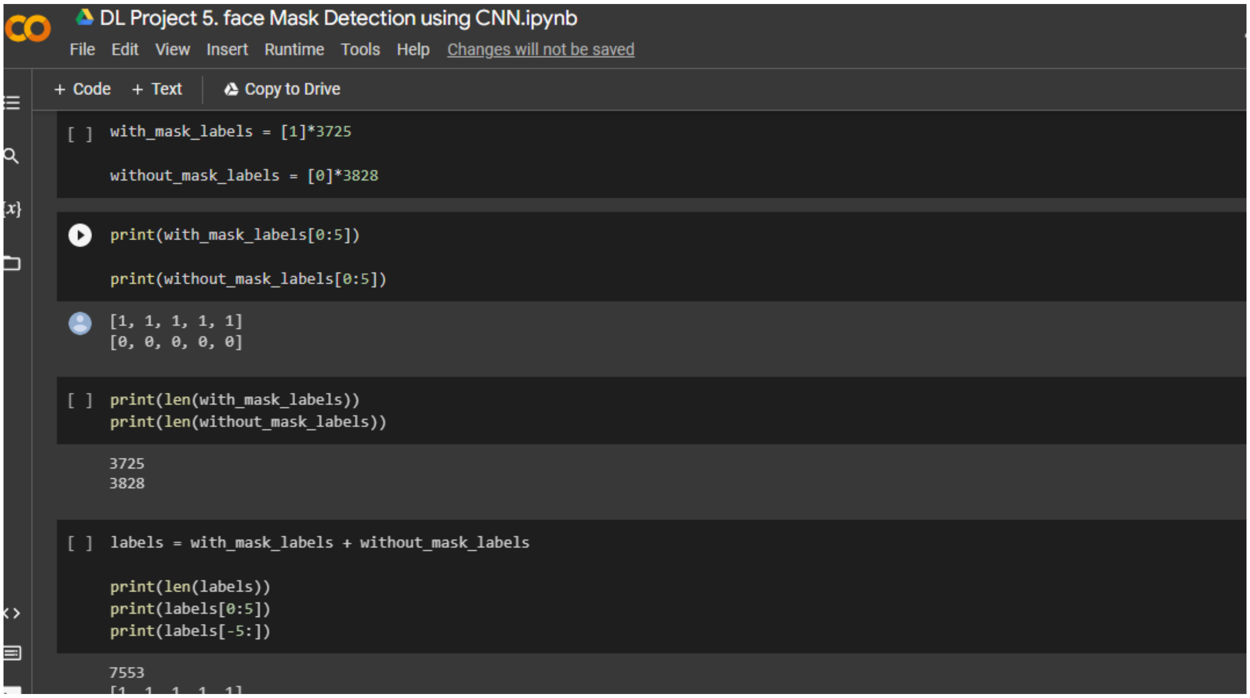
2.2. Data Validation Procedure:

The classification of a supervised learning CNN model is done after its training to classify the trained images to their respective classes by learning important visual patterns. TensorFlow and Keras are the primary building blocks for the proposed model. In this study, 80% of the dataset contributes to the training set and the rest to the testing set. The input image is pre-processed and augmented using the steps described above. There is a total of 5 Conv2D layers with ReLu activation functions with a 3 x 3 filter and 5 Max-Pooling Layers with a filter size of 2 x 2. Flatten and Dense are used as the fully connected layers. The output layer uses softmax as its activation function. This results in 2,818,658 trainable parameters in this Convolutional Neural Network (see Table 1). Tables 2 and 3 demonstrate the training process that is implemented using SGD and Adam. It compares them on different parameters such as Accuracy, Loss, Validation Accuracy, and Validation Loss. From Table 1 and Table 2, it is observed that Adam Optimizer works better than SGD optimizer as it provides better results with an increase in epochs. Figures 3 and 4 shows Adam optimizer gives better performance than SGD optimizer in all recall levels. The hyper-parameters used in this model are described below in Table 4. Binary_crossentropy is used for calculation of the classification loss for the model. For a classification problem, it yields a value between 0 and 1 (probability value).

2.3. Data Processing and Normalization:

The accuracy of a model is dependent on the quality of the dataset. The initial data cleaning is done to eliminate the faulty pictures discovered in the dataset. The images are resized into a fixed size of 96 x 96, which helps to reduce the load on the machine while training and to provide optimum results. The images are then labeled as being with or without masks. The array of images is then transformed to a NumPy array for quicker computation. Along with that, the preprocess input function from the MobileNetV2 is also used. Following that, the data augmentation technique is utilized to increase the quantity of training dataset and also improve it's quality. A function Image Data Generator is used with appropriate values of rotation, zoom, horizontal or vertical flip, to generate numerous versions of the same picture. The training samples have been increased to elude over-fitting. It enhances generalization and robustness of the trained model. The whole dataset is then divided into training data and test data in a ratio of 8:2 by randomly selecting images from the dataset. The stratify parameter is used to keep the same proportion of data as in the original dataset in both the training and testing datasets. Also Fig 1 is explained process.

2.4. Feature Extraction Procedure:



2.5. Classification Algorithms:

The model architecture adopted for the research is described in Table 1. The main components of the architecture are 2D convolutional layers (conv2D), pooling layer, activation functions and fully-connected layers. The proposed model comprises of a total of 5 Conv2D layers with padding ‘same’ and stride of 1. At each conv2D layer, feature map of 2D input data is extracted by “sliding input” across a filter or kernel and perform following operation:

$$C(Z) = (P \times Q)(x) = \left(\int_{-\infty}^{\infty} P(Z) \times Q(Z - x)\mathrm{d}Z\right.$$

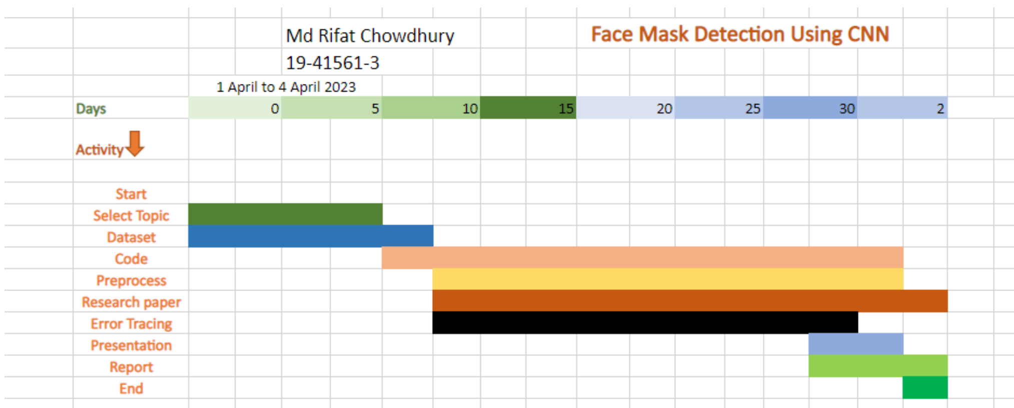
In the above, P represents the matrix of the input image, and Q is convolutional kernel giving C as output.
Pooling layers decrease the size of the feature map. Thus, the number of trainable parameters is reduced, resulting in rapid calculations without losing essential features. Two major kinds of pooling operations can be carried out: max pooling and average pooling. Max pooling implies making the most significant value present in the specific location where the kernel resides. On the other hand, average pooling computes the mean of every value in that region.
Activation functions are the nodes that are placed at the end or among neuronal networks (layers). They decide whether or not the neuron fires. Choice of activation function at hidden layers as well as at output layer is important as it controls the quality of model learning. The ReLU activation function is primarily used for hidden layers; whereas, Softmax is used for the output layer and calculates probability distribution from a real number vector. The latter is the

preferred choice for multi-class classification problems. Regarding ReLU, it offers better performance and widespread depth learning compared to the function of sigmoid and tanh [18]. After all Convolutional layers have been implemented, the FC layers are applied. These layers help to classify pictures in both the multi class and binary categories. In these layers, the softmax activation function is the choice of preference to produce probabilistic results.

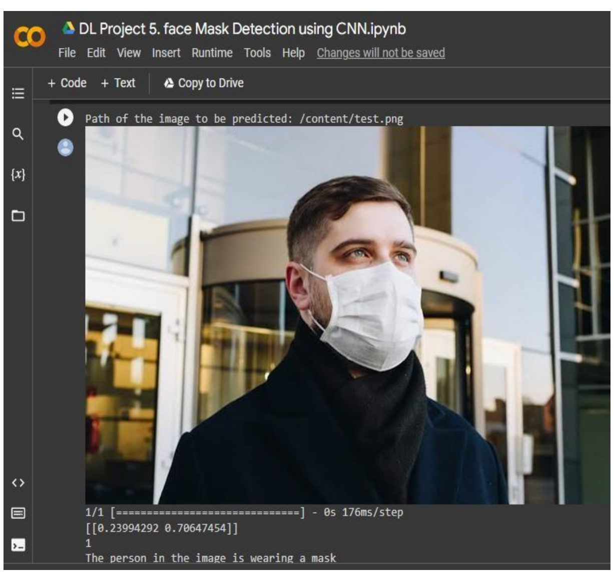
2.6. Data Analysis Techniques:

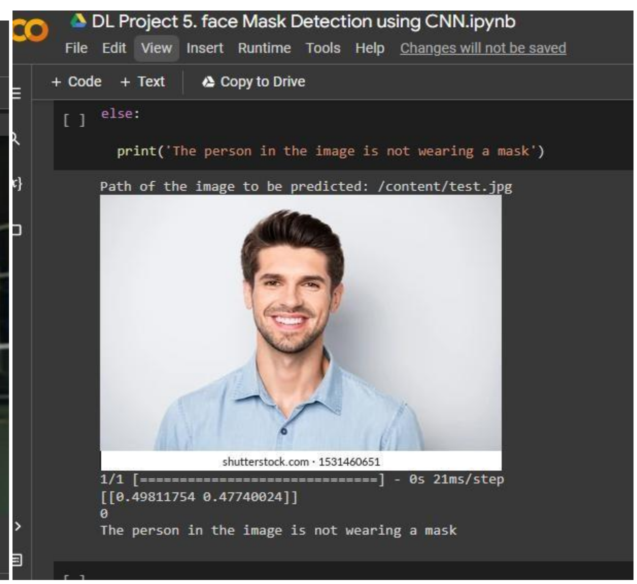
In the last step the CNN model is integrated into a web-based application that is hosted in order to be shared easily with other users and they can upload their image or live video feed to the model to recognize facial masks and then get the predicted result. Streamlit, an open-source python library, is used to design and construct a simple web app that allows users to submit a picture with a single click of a button and receive the outcome in a matter of seconds.

2.7. Block Diagram and Workflow Diagram of Proposed Model:



3. Results:

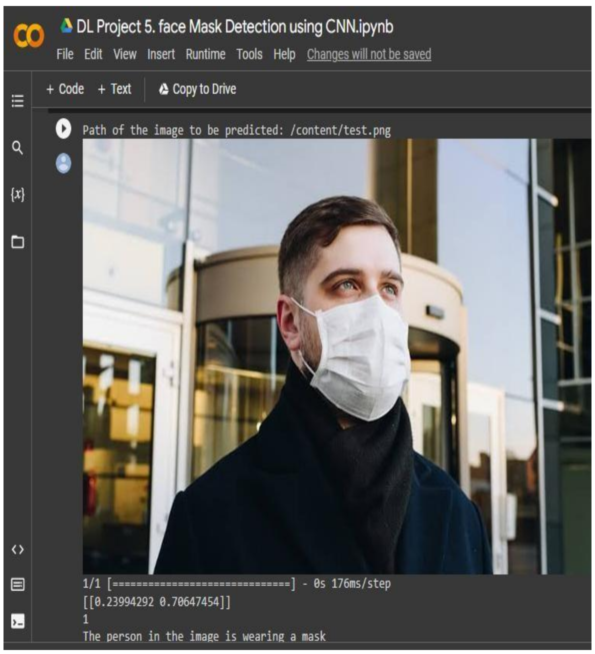


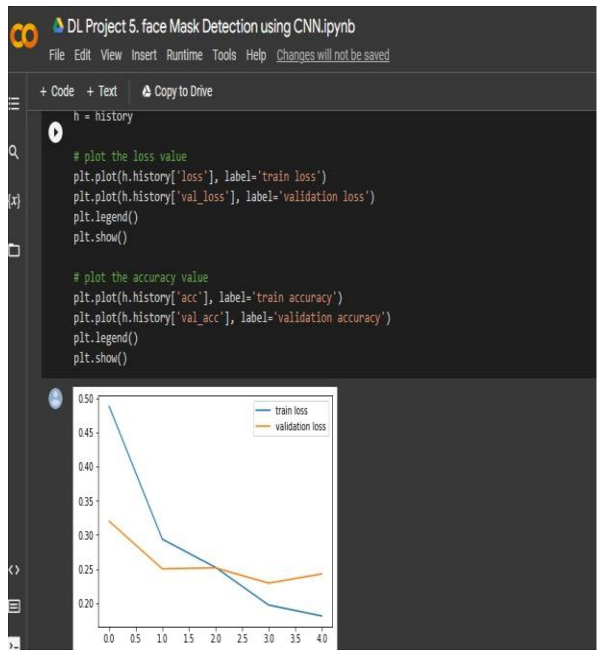


3.1. Results Comparison:

True positives are accurately predicted as being in a positive class, whereas false positives are images that were incorrectly predicted as being in a positive class. True negatives are accurately predicted to be in the negative class, whereas false negatives are incorrectly predicted to be in the negative class.
The accuracy of the masked individual identified by the developed model provides a good standard of prediction. Pictures above, which shows training accuracy, the model had struggled to acquire features until it reached 40 epochs, following which the curve remained steady. The accuracy was around 92% after 100 epochs. The green curve displays the training accuracy, while the blue line gives the validation dataset. Moreover, Results which shows the training and validation loss curve, the green line represents loss in the training dataset, which is smaller than 0.1, and the loss in the validation dataset is represented by the blue line, which is also less than
0.2 after 100 epochs.

3.2. Graphical Representation of Results:





3.4. Discussion:

The result of 1st image we can see the accuracy of increasing at the start and loss is seen decreasing after it. The accuracy reached 70%. So, the process of training into the deep neural network was much faster than the expectation. Our proposed model was able to classify the mask-wearing image with faces. The accuracy provided by the model in such a scenario was near about perfect as the model detected all the faces that had mask on them accurately along with the accuracy level. The result of 2nd data accurately detected the faces that weren't wearing masks, and the results at the current level are satisfactory.

4. Conclusion:

By using deep learning model for face mask detection for the prevention of COVID-19. Our system classifies if a person is wearing a face mask properly, wearing incorrectly or not wearing at all. The model we built has higher accuracy with training almost 7553 images. This research work can be crucial for curbing the spread of COVID-19 in society. Also, our model hence can be integrated in several applications for successful scalability and real-world impact.

5. Future Recommendations:

The proposed technique can be integrated into any high-resolution video surveillance devices and not limited to mask detection only. Secondly, the model can be extended to detect facial landmarks with a facemask for biometric purposes. We faced some issues like varying angles and lack of clarity of frame. We aim to improve these in future editions of the work.

6. References:

1. D. Chen, G. Hua, F. Wen and J. Sun, "Supervised transformer network for efficient face detection," 2016 European Conference on Computer Vision, Amsterdam, Netherlands, 2016, pp. 122-138, doi: 10.1007/978-3-319-46454-1_8.
2. Y. Chen, M. Hu, H. Cao, G. Zhai, J. Zhang, Q. Li and S. X. Yang, "Face mask assistant: detection of face mask service stage based on mobile phone," IEEE Sensors Journal, vol. 21, no. 9, pp. 11084-11093, 2021, doi: 10.1109/JSEN.2021.3061178.
3. A. P. Cruz and J. Jaiswal, "Text-to-image classification using attngan with densenet architecture," in Proceedings of International Conference on Innovations in Software Architecture and Computational Systems, Springer, 2021, pp. 1-17.
4. M. S. Ejaz, M. R. Islam, M. Sifatullah and A. Sarker, "Implementation of principal component analysis on masked and non-masked face recognition," in 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), IEEE, 2019, pp. 1-5, doi: 10.1109/ICASERT.2019.8934543.
5. K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.
6. A. Altun, F. Celebi and Y. Kaya, "A survey of image forgery detection techniques," IEEE Access, vol. 7, pp. 127837-127855, 2019, doi: 10.1109/ACCESS.2019.2934852.
7. S. Arif, S. Samiullah and M. Hussain, "Smart parking system based on internet of things," in 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, USA, 2018, pp. 2590-2593, doi: 10.1109/EMBC.2018.8512750.
8. A. Basu, B. Maiti, P. Dutta and A. Basu, "Role of machine learning in the field of agriculture," in 2020 International Conference on Smart Grid and Renewable Energy (ICSGRE), IEEE, 2020, pp. 110-114, doi: 10.1109/ICSGRC49013.2020.9232610.
9. J. Tang, H. Li, J. Li and Y. Chen, "Multi-scale feature fusion network for pavement crack detection," Sensors, vol. 20, no. 18, p. 5236, 2020, doi: 10.3390/s20185236.