# Deep Learning CNN based Robust Face Mask Detection

Prasanna Shivva, Gouni Akila

Department of Computer Science and Engineering

Sree Dattha Group of Institutions, Hyderabad, Telangana, India.

**Abstract**

COVID-19 epidemic has swiftly disrupted our day-to-day lives affecting the international trade and movements. Wearing a face mask to protect one's face has become the new normal. Soon, many public service providers will expect the clients to wear masks appropriately to partake of their services. Therefore, face mask detection has become a critical duty to aid worldwide civilization. This paper provides a simple way to achieve this objective utilising some fundamental Machine Learning tools as TensorFlow, Keras, OpenCV and Scikit-Learn. The suggested technique successfully recognises the face in the image or video and then determines whether it has a mask on it. As a surveillance job performer, it can also recognise a face together with a mask in motion as well as in a video. The technique attains excellent accuracy. We investigate optimal parameter values for the Convolutional Neural Network model (CNN) to identify the existence of masks accurately without generating over-fitting.

**Keywords:** Face mask detection, deep learning CNN, machine learning.

## 1. Introduction

### Overview

The spread of COVID-19 has resulted in more than 1,841,000 global deaths and more than 351,000 deaths in the US by Dec. 31, 2020. The spread of virus can be avoided by mitigating the effect of the virus in the environment or preventing the virus transfer from person to person by practicing physical distance and wearing face masks. WHO defined physical distancing as keeping at least 6-ft. or 2-m distance from others and recommended that keeping the physical distance and wearing a face mask can significantly reduce transmission of the COVID-19 virus. Like other sectors, the construction industry has been affected, where unnecessary projects have been suspended or mitigated people's interaction. However, many infrastructure projects cannot be suspended due to their crucial role in people's life. Therefore, bridge maintenance, street widening, highway rehabilitation, and other essential infrastructure projects have been activated again to keep the transportation system's serviceability. Although infrastructure projects are activated, the safety of construction workers cannot be overlooked. Due to the high density of workers in construction projects, there is a high risk of the infection spread in construction sites. Therefore, systematic safety monitoring in infrastructure projects that ensure maintaining the physical distance and wearing face masks can enhance construction workers' safety.

Safety agents are sometimes deployed to infrastructure projects to inspect workers to see whether they are complying with social distancing or wearing face masks. However, once there are so many workers on a construction site, it is difficult for the officers to determine hazardous situations. In addition, assigning safety officers increases the number of people on-site, raising the chance of transmission even more, and putting workers and officers in a more dangerous situation. Recently, online video capturing in construction sites has become very common. Drones are used in construction projects to record online videos to manage worksites more efficiently. The current system of online video capturing can be used for safety purposes. An automatic system that uses computer vision techniques to capture real-time safety violations from online videos can enhance infrastructure project workers' safety. This study develops a model using Faster R-CNN to detect workers who either do not wear a face mask or do not maintain the physical distance in road projects. Once a safety violation occurs, the model highlights who violates the safety rules by a red box in the video.

Taneja et. al [1] proposed and implemented a face mask detection model that can accurately detect whether a person is wearing a mask or not. The model architecture uses MobileNetV2, which is a lightweight convolutional neural network, therefore requires less computational power and can be easily embedded in computer vision systems and mobile. As a result, it can create a low-cost mask detector system that can help to identify whether a person is wearing a mask or not and act as a surveillance system as it works for both real-time images and videos. The face detector model achieved high accuracy of 99.98% on training data, 99.56% on validation data, and 99.75% on testing data. Mohamed Loey et. al [2] presented a hybrid model using deep and classical machine learning for face mask detection. The proposed model consists of two components. The first component is designed for feature extraction using Resnet50. While the second component is designed for the classification process of face masks using decision trees, Support Vector Machine (SVM), and ensemble algorithm. Three face masked datasets have been selected for investigation. The Three datasets are the Real-World Masked Face Dataset (RMFD), the Simulated Masked Face Dataset (SMFD), and the Labeled Faces in the Wild (LFW). The SVM classifier achieved 99.64% testing accuracy in RMFD. In SMFD, it achieved 99.49%, while in LFW, it achieved 100% testing accuracy.

Nowrin et. al [3] presented a narrative and meta-analytic review covering all the existing facemask detection algorithms, considering the context of Covid-19. The procedure of the existing algorithms, their considerations, effectiveness, evaluation process, and outcomes were presented. Moreover, the datasets used in those algorithms were discussed briefly. The shortcomings of the existing algorithms were reviewed, and the future challenges were outlined. Although a significant amount of research has been focused on developing an efficient facemask detection algorithm, they mainly concentrated on the same set of problems neglecting some other significant issues. This paper highlighted those shortcomings, such as, maintaining image-resolution during detection process, scarcity of rich dataset, categorical classifications, and others. Also, it specified the future scopes which includes diversity in datasets and facemask types, different facemask wearing conditions, reconstruction of the masked face, and so on. This comprehensive review will pave the way for the research community to understand the current facemask detection algorithms. By analyzing the shortcomings and future challenges in this field, researchers will develop novel approaches to fill those gaps.

## 2. Literature survey

Peishu Wu et. al [4] proposed an efficient automatic face mask recognition and detection framework FMD-Yolo and corresponding algorithm. In particular, a modified Res2Net structure Im-Res2Net-101 serves as the backbone to extract features with rich receptive fields from the input. Subsequently there follows a feature fusion component En-PAN, which is a novel path aggregation network and primarily consists of Yolo Residual Block, SPP, Coord Conv, SE Layer blocks.

Singh et. al [5] proposed an efficient real-time deep learning-based technique to automate the process of detecting masked faces, where each masked face is identified in real-time with the help of bounding boxes. The extensive trials were conducted with popular models, namely, Faster RCNN and YOLO v3. F-RCNN has better precision, but for applying this in real-world surveillance cameras, it would be preferred to use the model with YOLO algorithm as it performs single-shot detection and

ISSN- 2394-5125 VOL 07, ISSUE 10, 2020

has a much higher frame rate than Faster-RCNN or any other state-of-the-art object detection algorithms.

Mohamed Loey et. al [6] introduced a novel model for medical masked face detection, focusing on medical mask object to prevent COVID-19 spreads from human to human. For image detection, they have employed the YOLO v2 based ResNet-50 model to produce high-performance outcomes. The proposed model improves detection performance by introducing mean IoU to estimate the best number of anchor boxes. To train and validate our detector in a supervised state, they design a new dataset based on two public masked face datasets.

Zhang et. al [7] first propose a practical and challenging dataset, which aims to reflect the conditions of wearing face mask in the era of COVID-19. Then we analyze the main challenging points in this task. Based on these analyses, we further develop Context-Attention R-CNN, a framework to detect conditions of wearing face mask, which contains three novel points: multiple context feature extractor, decoupling branches and attention module. With these components, the Context-Attention R-CNN brings significant improvements for region-based detector. The extensive experiments show that Context-Attention R-CNN outperforms many state-of-the-art detectors, including two-stage detectors and single-stage detectors.

Wang et. al [8] proposed a hybrid deep transfer learning and BLS for facial mask detection. It is designed to contain two stages: predetection and verification. The predetection is implemented by the Faster_RCNN framework through a transfer learning technique. The detection model is fine-tuned from a multiple-class detection model. The verification is implemented by a classifier of BLS. With a low score setting in predetection, more candidate regions are used for verification

Fan et. al [9] proposed a novel SL-FMDet, which is efficient and has low hardware requirements. To overcome the lower feature extraction capability caused by its light-weight backbone, we proposed RCAM and SGHR. RCAM can extract rich context information and focus on crucial face mask related areas. By using SGHR as an auxiliary task, the model is able to learn more discriminating features for faces with and without masks. The model with SGHR yielded a better attention map, which qualitatively supports the effectiveness of this auxiliary task.

Agarwal et. al [10] proposed an intelligent face mask detector framework based on deep learning concept which can classify the person who wear mask from those who are not wearing mask. In the proposed work, a hybrid model of convolution neural network with support vector machine is used for designing the mask detector. The performance of the proposed method is evaluated on real-world masked face recognition dataset (RMFD) and medical mask dataset (MDD). When implemented, it has been found that the proposed method can achieve high accuracy (99.11%). The excellent performance of the proposed model is very suitable for video surveillance equipment also.

Razavi et. al [11] developed a computer vision system to automatically detect the violation of face mask wearing and physical distancing among construction workers to assure their safety on infrastructure projects during the pandemic. For the face mask detection, we collected and annotated 1000 images, including different types of face mask wearing, and added them to a pre-existing face mask dataset to develop a dataset of 1853 images and increased the dataset to 3300 images by data augmentation.

## 3. Proposed system

### Face Mask Dataset

Masks play a crucial role in protecting the health of individuals against respiratory diseases, as is one of the few precautions available for COVID-19 in the absence of immunization. With this dataset, it is possible to create a model to detect people wearing masks, not wearing them, or wearing masks improperly.

This dataset contains 853 images belonging to the 3 classes, as well as their bounding boxes in the PASCAL VOC format.

The classes are

- With mask.
- Without mask.

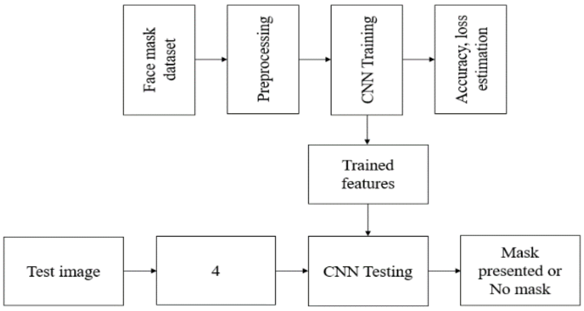



Fig. 1: Block diagram of proposed system.

**Dataset Description**

In this project we are detecting whether person has worn a mask or not and to implement this project we have used following packages

Opencv: using this package we can read and write images and can able to detect faces from images

Keras and tensorflow: tensorflow provide libraries to build CNN model and using keras we can define layers for that CNN model

To implement this project, we have used dataset from below link

https://github.com/prajnasb/observations/tree/master/experiements/data

**Data Pre-processing in Machine learning**

Data pre-processing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a machine learning project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data pre-processing task.

**Why do we need Data Pre-processing?**

ISSN- 2394-5125 VOL 07, ISSUE 10, 2020

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data pre-processing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

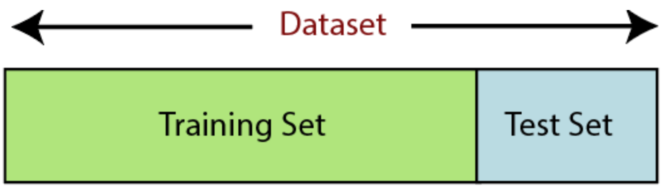
- Getting the dataset
- Importing libraries
- Importing datasets
- Finding Missing Data
- Encoding Categorical Data
- Splitting dataset into training and test set
- Feature scaling

**Splitting the Dataset into the Training set and Test set**

In machine learning data pre-processing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model.

Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:




**Training Set**: A subset of dataset to train the machine learning model, and we already know the output.

**Test set**: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

**Proposed ResNet-CNN**

Deep neural network is gradually applied to the identification of crop diseases and insect pests. Deep neural network is designed by imitating the structure of biological neural

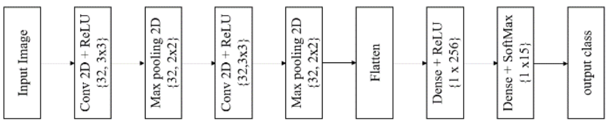



Fig. 2: Proposed ResNet-CNN.

network, an artificial neural network to imitate the brain, using learnable parameters to replace the links between neurons. Convolutional neural network is one of the most widely used deep neural network structures, which is a branch of feed forward neural network. The success of AlexNet network model also confirms the importance of convolutional neural network model. Since then, convolutional neural networks have developed vigorously and have been widely used in financial supervision, text and speech recognition, smart home, medical diagnosis, and other fields.

Convolutional neural networks are generally composed of three parts. Convolution layer for feature extraction. The convergence layer, also known as the pooling layer, is mainly used for feature selection. The number of parameters is reduced by reducing the number of features. The full connection layer carries out the summary and output of the characteristics. A convolution layer is consisting of a convolution process and a nonlinear activation function ReLU. A typical architecture of CNN model for crop disease recognition is shown in Fig. 11.

The leftmost image is the input layer, which the computer understands as the input of several matrices. Next is the convolution layer, the activation function of which uses ReLU. The pooling layer has no activation function. The combination of convolution and pooling layers can be constructed many times. The combination of convolution layer and convolution layer or convolution layer and pool layer can be very flexibly, which is not limited when constructing the model. But the most common CNN is a combination of several convolution layers and pooling layers. Finally, there is a full connection layer, which acts as a classifier and maps the learned feature representation to the sample label space.

**ResNet-CNN**

According to the facts, training and testing of ResNet-CNN involves in allowing every source image via a succession of convolution layers by a kernel or filter, rectified linear unit (ReLU), max pooling, fully connected layer and utilize SoftMax layer with classification layer to categorize the objects with probabilistic values ranging from [0,1]. Figure 1 discloses the architecture of ResNet-CNN that is utilized in proposed methodology for CBIR system for enhanced feature representation of word image over conventional retrieval systems.

Convolution layer as depicted is the primary layer to extract the features from a source image and maintains the relationship between pixels by learning the features of image by employing tiny blocks of source data. It's a mathematical function which considers two inputs like source image $I(x, y, d)$ where $x$ and $y$ denotes the spatial coordinates i.e., number of rows and columns. $d$ is denoted as dimension of an image (here $d = 3$, since the source image is RGB) and a filter or kernel with similar size of input image and can be denoted as $F(k_x, k_y, d)$.

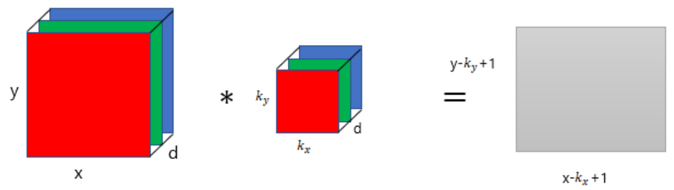
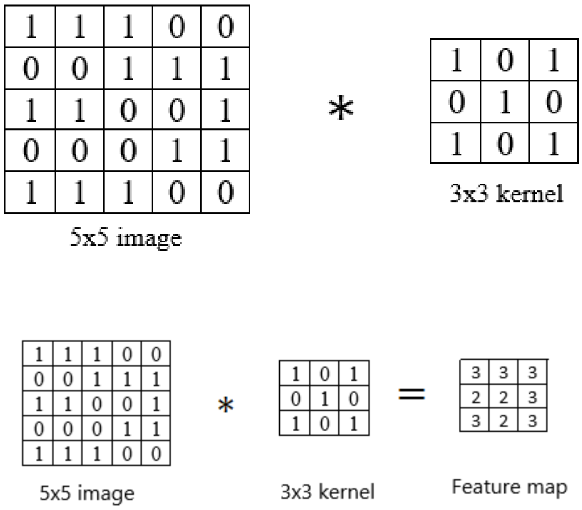



Fig. 3: Representation of convolution layer process.

The output obtained from convolution process of input image and filter has a size of $C\left((x - k_x + 1), (y - k_y + 1), 1\right)$, which is referred as feature map. Let us assume an input image with a size of $5 \times 5$ and the filter having the size of $3 \times 3$. The feature map of input image is obtained by multiplying the input image values with the filter values as given in Fig. 12.

ISSN- 2394-5125 VOL 07, ISSUE 10, 2020




(

(

Fig. 4: Example of convolution layer process.

**ReLU layer**

Networks those utilizes the rectifier operation for the hidden layers are cited as rectified linear unit (ReLU). This ReLU function $\mathcal{G}(\cdot)$ is a simple computation that returns the value given as input directly if the value of input is greater than zero else returns zero. This can be represented as mathematically using the function $max(\cdot)$ over the set of 0 and the input $x$ as follows:

$$\mathcal{G}(x) = \max\{0, x\}$$

**Max pooing layer**

This layer mitigates the number of parameters when there are larger size images. This can be called as subsampling or down sampling that mitigates the dimensionality of every feature map by preserving the important information. Max pooling considers the maximum element form the rectified feature map.

**Softmax classifier**

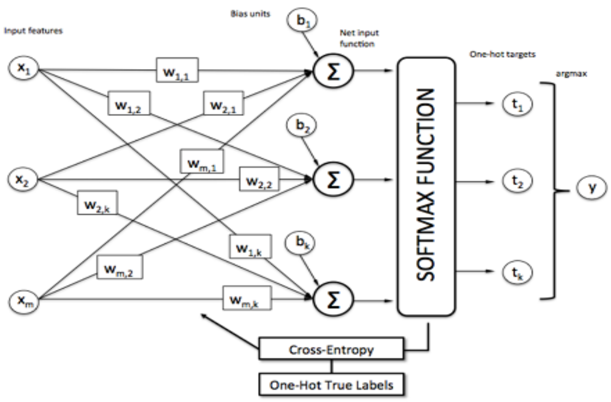



Fig. 5: Crop disease prediction using SoftMax classifier.

Generally, as seen in the above picture softmax function is added at the end of the output since it is the place where the nodes are meet finally and thus, they can be classified. Here, X is the input of all the models and the layers between X and Y are the hidden layers and the data is passed from X to all the layers and Received by Y. Suppose, we have 10 classes, and we predict for which class the given input belongs to. So, for this what we do is allot each class with a particular predicted output. Which means that we have 10 outputs corresponding to 10 different class and predict the class by the highest probability it has.

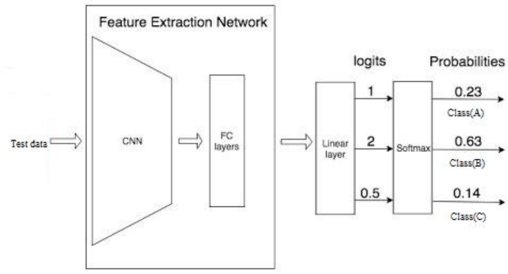



Fig. 6: Example of SoftMax classifier.

In Fig. 4.6, and we must predict what is the object that is present in the picture. In the normal case, we predict whether the crop is A. But in this case, we must predict what is the object that is present in the picture. This is the place where softmax comes in handy. As the model is already trained on some data. So, as soon as the picture is given, the model processes the pictures, send it to the hidden layers and then finally send to softmax for classifying the picture. The softmax uses a One-Hot encoding Technique to calculate the cross-entropy loss and get the max. One-Hot Encoding is the technique that is used to categorize the data. In the previous example, if softmax predicts that the object is class A then the One-Hot Encoding for:

Class A will be [1 0 0]

Class B will be [0 1 0]

Class C will be [0 0 1]

From the diagram, we see that the predictions are occurred. But generally, we don't know the predictions. But the machine must choose the correct predicted object. So, for machine to identify an object correctly, it uses a function called cross-entropy function.

So, we choose more similar value by using the below cross-entropy formula.

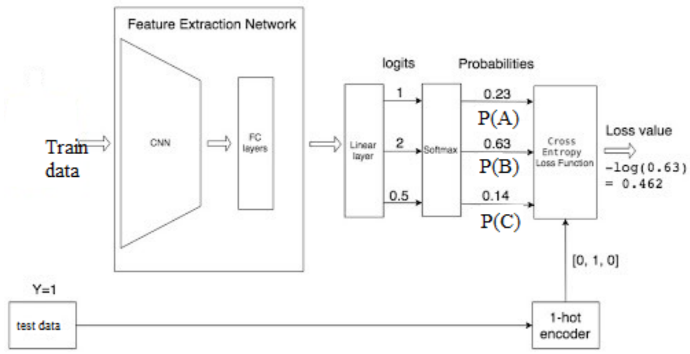



Fig. 7: Example of SoftMax classifier with test data.

In the above example we see that 0.462 is the loss of the function for class specific classifier. In the same way, we find loss for remaining classifiers. The lowest the loss function, the better the prediction is. The mathematical representation for loss function can be represented as: -

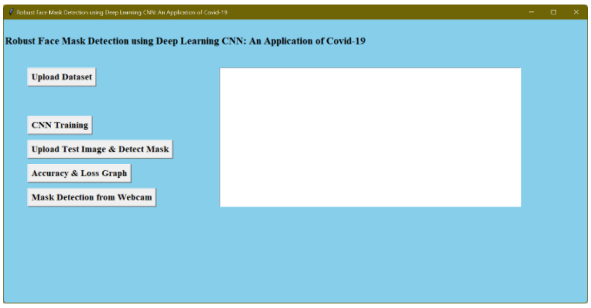
$$LOSS = np.sum(-Y \ * \ np.log(Y\_pred))$$

ISSN- 2394-5125 VOL 07, ISSUE 10, 2020

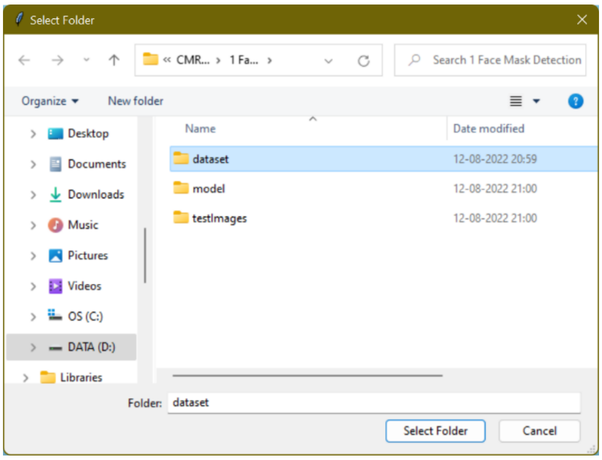
## 4. Results

### Modules

- Upload Mask Detection Dataset: This user will upload the Mask Detection Dataset and then the dataset may load.
- Train Mask Images using CNN: This is the second module in our project, after loading the dataset we can train with CNN.
- Upload Test Image & Detect Mask: Then we must upload our test image to detect mask.
- Accuracy & Loss Graph: In graph x-axis represents EPOCH and y-axis represents accuracy and loss values.

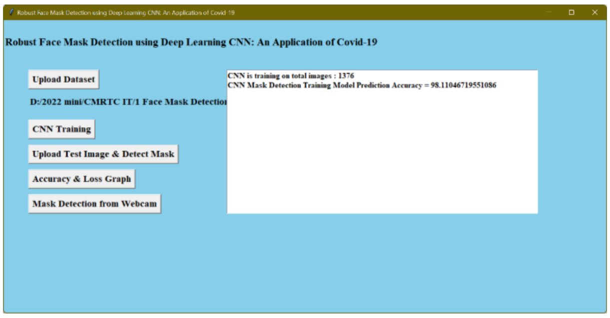



In above screen click on 'Upload Dataset' button to upload dataset and to get below screen

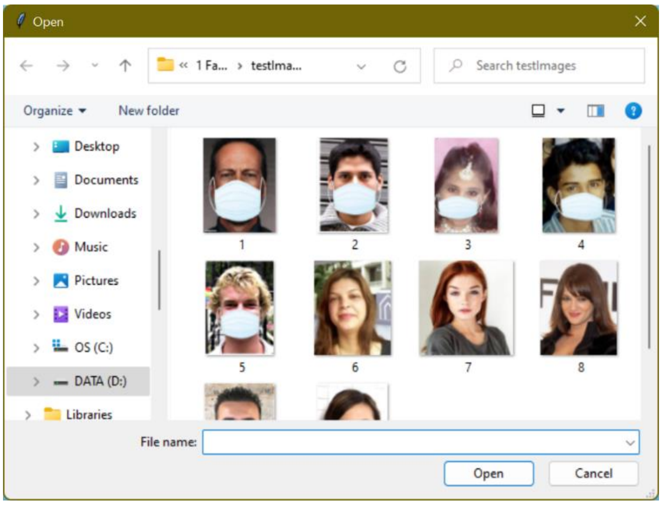



In above screen selecting and uploading 'dataset' folder and then click on 'Select Folder' button to load dataset.

Now click on 'CNN Training' button to train CNN with loaded dataset and while building CNN application will read images and then pre-process images and then build model.



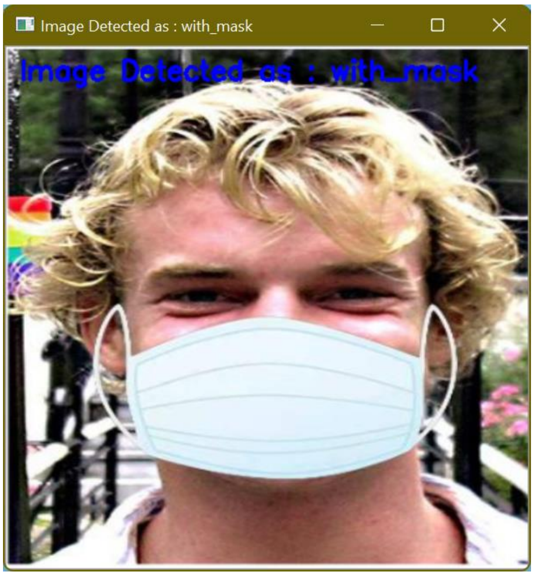

In above screen we can see CNN trained on 1376 images and its prediction accuracy is 98% ad now model is ready and now click on ‘Upload Test Images & Detect Mask’ button to upload test image and then application predict whether person wear mask or not.



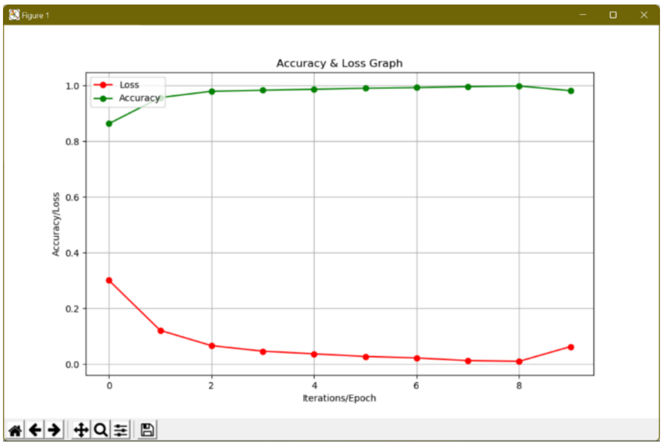

In above screen selecting and uploading 7.jpg file and then click on ‘Open’ button to get below screen


In above screen application saying image is without mask and you can see predicted result in image title bar also. Now test with other images

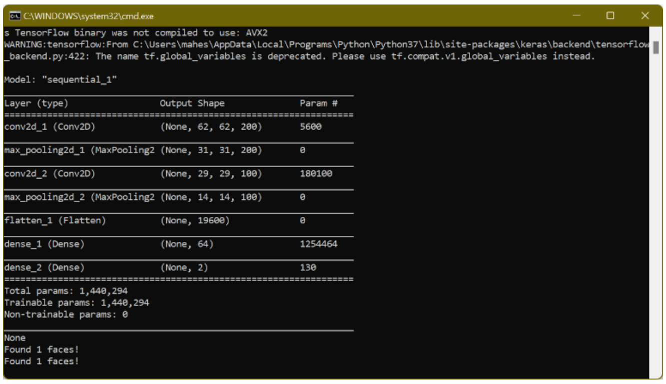
ISSN- 2394-5125 VOL 07, ISSUE 10, 2020




Similarly, you can upload any image and application will perform detection. Now click on 'Accuracy and Loss Graph' button to get below graph.




In above screen red refers to loss and green line refers to accuracy and x-axis represents EPOCH and y-axis represents accuracy and loss value. In above graph with each increasing epoch accuracy get better and loss get decrease. Below screen showing CNN model with 200, 100 layers for input and 64 for output.



```
C:\WINDOWS\system32\cmd.exe
s TensorFlow binary was not compiled to use: AVX2
WARNING:tensorflow:From C:\Users\mahes\AppData\Local\Programs\Python\Python37\lib\site-packages\keras\backend\tensorflow
_backend.py:422: The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 62, 62, 200)       5600
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 31, 31, 200)       0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 29, 29, 100)       180100
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 14, 14, 100)       0
_________________________________________________________________
flatten_1 (Flatten)          (None, 19600)             0
_________________________________________________________________
dense_1 (Dense)              (None, 64)                1254464
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 130
=================================================================
Total params: 1,440,294
Trainable params: 1,440,294
Non-trainable params: 0
_________________________________________________________________
None
Found 1 faces!
Found 1 faces!
```

In above screen multiple layers created where 62 X 62 is the image size and 200 is the filters for first layer. In below screen we are showing output without face


In above screen application saying face not detected in uploaded image.

## 5. Conclusion

Measures must be taken to control the spread of the COVID19 pandemic. This face mask recognition system is a very good and efficient way to do so. The system will separate the people from the crowd who are not wearing mask. The identification of people, violating the COVID norms increases the adaptability of the face mask detection system for the public sake. If applied in a correct way, the face mask detection system could be used to make sure our safety and for others too. This approach gives not only helps in achieving high precision but also enhance the face detection tempo considerably. The system can be applied in many areas like metro stations, markets, schools, railway stations and many other crowded places to monitor the crowd and to ensure that everyone is wearing mask.

### 5.1 Future scope

This work can be used for future researchers and enthusiasts. Firstly, this model can be used in any high-definition camcorders, this will make sure that this model is not limited to only face mask detection system. Secondly, this can be used for biometric scans with a mask on the face.

## References

[1] Taneja, S., Nayyar, A., Vividha, Nagrath, P. (2021). Face Mask Detection Using Deep Learning During COVID-19. In: Singh, P.K., Wierzchoń, S.T., Tanwar, S., Ganzha, M., Rodrigues, J.J.P.C. (eds) Proceedings of Second International Conference on Computing, Communications, and Cyber-Security. Lecture Notes in Networks and Systems, vol 203. Springer, Singapore. https://doi.org/10.1007/978-981-16-0733-2_3

[2] Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N. Taha, Nour Eldeen 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, Volume 167, 2021, 108288, ISSN 0263-2241, https://doi.org/10.1016/j.measurement.2020.108288.

[3] A. Nowrin, S. Afroz, M. S. Rahman, I. Mahmud and Y. -Z. Cho, "Comprehensive Review on Facemask Detection Techniques in the Context of Covid-19," in IEEE Access, vol. 9, pp. 106839-106864, 2021, doi: 10.1109/ACCESS.2021.3100070.

[4] Peishu Wu, Han Li, Nianyin Zeng, Fengping Li, FMD-Yolo: An efficient face mask detection method for COVID-19 prevention and control in public, Image and Vision Computing, Volume 117, 2022, 104341, ISSN 0262-8856, https://doi.org/10.1016/j.imavis.2021.104341.

[5] Singh, S., Ahuja, U., Kumar, M. et al. Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment. Multimed Tools Appl 80, 19753–19768 (2021). https://doi.org/10.1007/s11042-021-10711-8

[6] Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N. Taha, Nour Eldeen 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, Volume 65, 2021, 102600, ISSN 2210-6707, https://doi.org/10.1016/j.scs.2020.102600.

[7] J. Zhang, F. Han, Y. Chun and W. Chen, "A Novel Detection Framework About Conditions of Wearing Face Mask for Helping Control the Spread of COVID-19," in IEEE Access, vol. 9, pp. 42975-42984, 2021, doi: 10.1109/ACCESS.2021.3066538.

[8] B. Wang, Y. Zhao and C. L. P. Chen, "Hybrid Transfer Learning and Broad Learning System for Wearing Mask Detection in the COVID-19 Era," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-12, 2021, Art no. 5009612, doi: 10.1109/TIM.2021.3069844.

[9] X. Fan, M. Jiang and H. Yan, "A Deep Learning Based Light-Weight Face Mask Detector with Residual Context Attention and Gaussian Heatmap to Fight Against COVID-19," in IEEE Access, vol. 9, pp. 96964-96974, 2021, doi: 10.1109/ACCESS.2021.3095191.

[10] Agarwal, C., Kaur, I., Yadav, S. (2023). Hybrid CNN-SVM Model for Face Mask Detector to Protect from COVID-19. In: Gupta, M., Ghatak, S., Gupta, A., Mukherjee, A.L. (eds) Artificial Intelligence on Medical Data. Lecture Notes in Computational Vision and Biomechanics, vol 37. Springer, Singapore. https://doi.org/10.1007/978-981-19-0151-5_35

[11] Razavi, M., Alikhani, H., Janfaza, V. et al. An Automatic System to Monitor the Physical Distance and Face Mask Wearing of Construction Workers in COVID-19 Pandemic. SN COMPUT. SCI. 3, 27 (2022). https://doi.org/10.1007/s42979-021-00894-0