Training Report

on

"Detection of Improperly Worn Face Masks: a preventive measure against the spread of COVID-19"

done at

Jaypee University of Information Technology, Solan

(In-house Training)

Period of Training: 21st May, 2020 – 30th June, 2020

done by

Name: Anubha Bhaik

Roll No.: 171366

Branch: Computer Science

Department of Computer Science and Engineering
Jaypee University of Information Technology, Solan.

CONTENT

ACKNOWLEDGEMENT

With immense pleasure, I present my project report titled "**Detection of Improperly Worn Face Masks: a preventive measure against the spread of COVID-19**" as a part of the curriculum of "Bachelor of Technology".

The in-house training opportunity I had with Jaypee University of Information Technology was a great chance for learning and enhancing my skills. I am grateful to the Computer Science Department of Jaypee University of Information Technology for providing me with this opportunity.

I express my profound thanks to Dr. Ekta Gandotra, my in-house training faculty coordinator, who constantly helped and guided me in the completion of this project.

Thanking you.

Anubha Bhaik

Date: 5th August, 2020

1. INTRODUCTION

Corona virus disease 2019 has had a pressing impact on people all around the world. Ceasing the spread of this infectious disease is the urgent need of the hour. A vital method of protection against the virus is wearing masks in public areas. Not merely wearing masks, but wearing masks properly can ensure that the respiratory droplets do not get transmitted to other people.

Face masks are being used by people all over the world now. In many countries, it is now compulsory to wear a face mask when stepping out of home. However, many people do not wear the face masks properly. They fidget with their masks and pull them under their noses or completely off their faces to rest under their chins without realizing that improperly wearing a mask leads to an increased risk of contamination. Wearing a face mask limits the spread of the virus from someone who knows or does not know they have an infection or not.

The main aim of the project was to develop a deep learning based system to detect whether the person is wearing a mask properly or not. A convolutional neural network model based on the concept of transfer learning was trained on a self-made dataset of images and implemented with a light-weighted neural network MobileNetV2. OpenCV is used with Caffe framework to select facial detections which are further used on our pre-trained CNN model for classification. The method has been implemented on various input images and classification results have been obtained for the same.

2. DATA COLLECTION and PRE-PROCESSING

The images of people wearing proper face masks are collected from the images present in existing datasets [1],[2] and various other sources on the Internet. Since these datasets are having less number of images of improperly worn masks, so such images are collected from the Internet and local lab. Finally the dataset consists of 500 images, equally distributed among properly and improperly worn masks categories. Figure 1 shows the sample images with proper and improper masks from the dataset.

Figure 1: Sample images of the dataset belonging to two classes - proper and improper

The dataset so created consists of images of different sizes and thus, these are converted into a uniform size of 224×224 pixels. After the application of RGB reshaping, a $224 \times 224 \times 3$ image is given as input to the proposed model. The class labels are one-hot encoded. These pre-processed images and encoded labels are added to separate lists, one for the pre-processed images and the other for the class labels. Further, data augmentation parameters like random rotation, shift, shear, zoom and flip, are applied on the images which help in improving the performance generalization of the model.

3. WORKFLOW OF THE PROJECT:

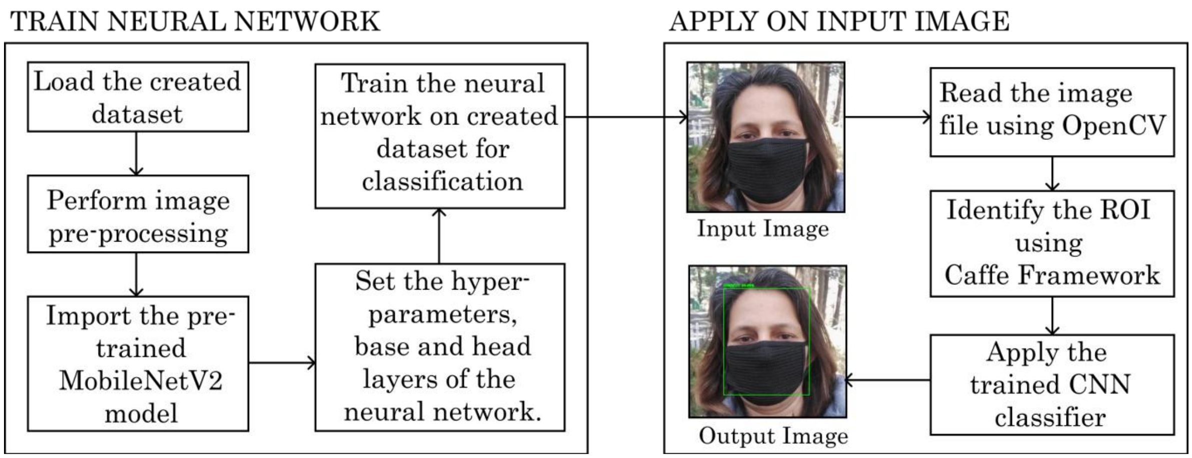


Figure 2: Workflow of the Project

3.1 Training the Neural Network

The dataset was split into training set and testing set in such a way that 80% data is used for the training purpose and 20% data is used for the testing purpose. Here, MobileNetV2 has been used as the base model [3]. The pre-trained weights for the ImageNet [4] dataset have been used as the backbone.

The last five custom layers which produce output for the model include the average pooling 2D layer with pool size 7×7. This layer is succeeded by a flatten layer that reshapes the pooled feature map to a single column vector. The simple feature vector is now put into a dense layer of 128 units of size accompanied by ReLU. A dropout layer is applied on this dense layer with threshold value of 0.5. Then a final dense layer is applied with Softmax non-linear activation to provide two output values, i.e., probability of the image belonging to the proper mask group and probability of the image belonging to the improper mask group. Adam optimizer is used for the optimization of the CNN and binary cross-entropy as loss function. This loss function is used in a binary classification problem. The initial learning rate is set to 0.0001.

3.2 Applying on Input Image (Detection of Region of Interest (ROI))

After training the CNN, Caffe, a deep learning framework, is used along with the Open Source Computer Vision library (OpenCV) for face detection using input images. For the purpose of extracting the Region of Interest (ROI) in the image, the DNN module of OpenCV is used with Caffe. The file with the Caffe framework format has been provided by the OpenCVfor face detection [5], [6] and it contains the weights for the actual layers. The Caffe model is based on the Single Shot MultiBox Detector (SSD) framework which uses ResNet as a base network for facial recognition as in [7].

An input image is first uploaded and pre-processed using OpenCV DNN module. The spatial dimensions of the input image are extracted and converted into a 4-D Binary Large OBject (BLOB). After normalizing the input image to create a BLOB, it is passed through the DNN to obtain face detections. The detections obtained are further checked for the probability or confidence which is used to classify the input image as proper or improper. The threshold confidence (or probability) is kept at 0.5 to filter all the weak detections. Further, OpenCV is used to extract the region of interest (ROI) of the face which helps in displaying the bounding box. The extracted face ROI is converted from BGR to RGB ordering of channels and the image size is set to 224×224 pixels to pass it through the trained model. Finally, this pre-processed input image is passed through the trained model to determine if the mask is worn correctly or not.

This can finally be visualized by a bounding box labelled with the class score in the image. The class score is the probability that the image contains a face with a proper or an improper mask.

4. EVALUATION RESULTS

Confusion matrix is calculated on the test set as shown in Table I.

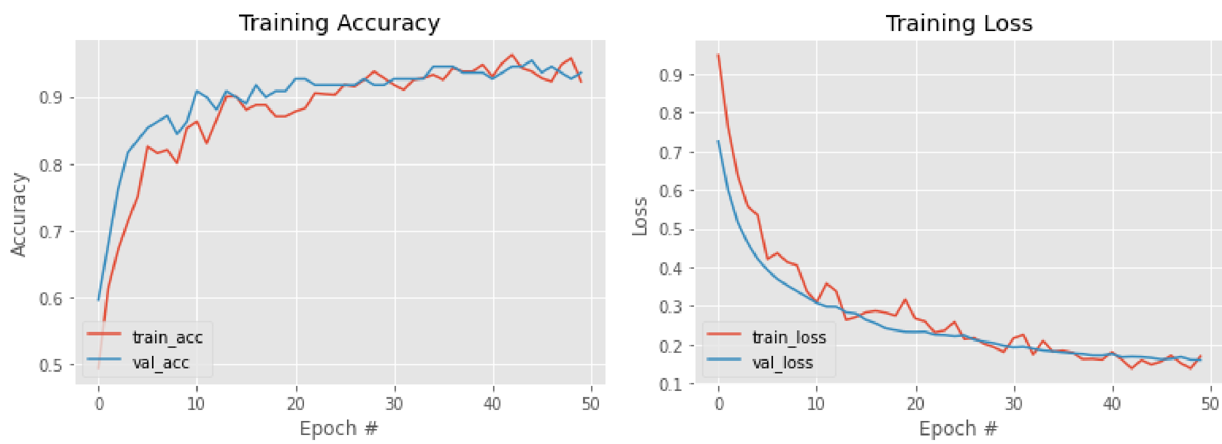
Table I: Confusion Matrix

		Predicted Classes	
		Proper	Improper
Actual Classes	Proper	TP = 47	FN = 3
	Improper	FP = 4	TN = 55

The other evaluation parameters calculated from the confusion matrix are presented in Table II.

Table II: Evaluation Metrics

	Evaluation Metric	Value (in %)
1.	Accuracy	93.58
2.	Precision	92.15
3.	Recall	94.00
4.	Sensitivity	94.00
5.	Specificity	93.22



(a) (b)

Figure 3: Learning Curves for Accuracy and Loss

Fig. 4 shows the plotted AUC of ROC curve of the proposed model which is above the threshold level and is calculated as 0.9361.

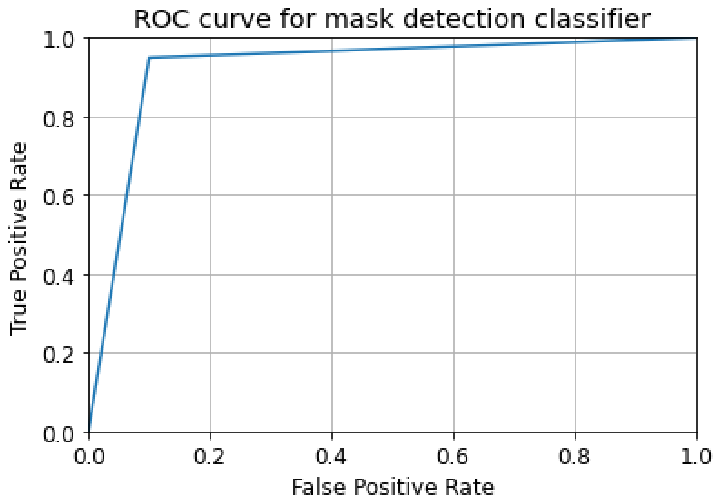


Figure 4: ROC Curve plotted on test set data

The images that were not used in training were provided as input to the proposed model to predict whether they are wearing the mask properly. Following are the results obtained on experimental images as shown in Figure 5.

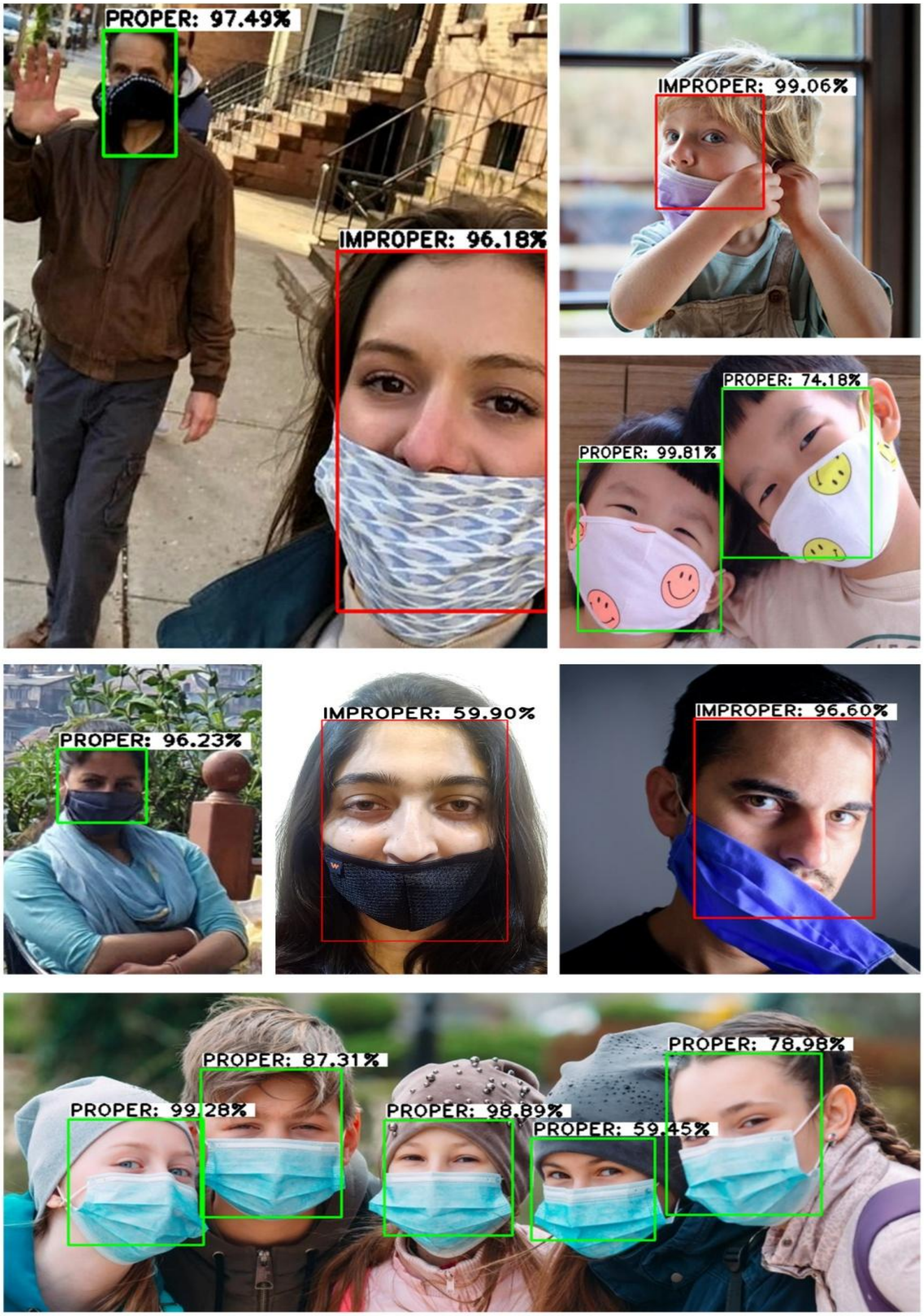


Figure 5: Output Images with predicted results as bounding box and label

5. CONCLUSION

The COVID-19 disease is the greatest challenge that the world has faced since World War II. To prevent its rapid spread, it is necessary that face masks are worn properly by people over their noses. People at crowded places, hospitals, offices and working spaces can be checked for improperly worn masks to ensure safety. Application of the proposed model can serve as a preventive measure in the COVID-19 crisis and benefit in safekeeping the health of society. The government can also leverage the model to detect improperly worn face masks in public places.

This project is specifically focused on classifying the mask worn by a person into two classes: proper and improper. This will be much significant in the various stages of unlocking all over the world as it will contribute to public safety and healthcare. The architecture of this model consists of the light weighted MobileNetV2 neural network as the backbone which overcomes computational issues. MobileNetV2 can efficiently be used on devices with low computational power as well. Transfer learning has been adopted to use weights that have been used for a similar task like face detection and already trained on a very large dataset. Furthermore, OpenCV with the Caffe framework has been used to detect facial features on experimental input images and used on the pre-trained model with our dataset, to produce classification results with indicative results, such as labels and a bounding box. A model accuracy of 93.58%, and an AUC measure of 0.936 was finally achieved.

REFERENCES

[1] H. Baojin, “Real-World Masked Face Dataset， RMFD.” Real-World Masked Face Dataset， RMFD.

[2] P. Bhandari, “Mask Classifier Dataset, 2020.” https://github.com/prajnasb/observations.

[3] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[4] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[5] “OpenCV github repository - How to train face detector.” https://github.com/opencv/opencv/blob/4.0.0-beta/samples/dnn/face_detector/how_to_train_face_detector.txt.

[6] “OpenCV github repository - Face Detector: deploy.prototxt.” https://github.com/opencv/opencv/blob/4.0.0-beta/samples/dnn/face_detector/deploy.prototxt.

[7] Y. Jia *et al.*, “Caffe: Convolutional Architecture for Fast Feature Embedding,” in *Proceedings of the 22nd ACM International Conference on Multimedia*, 2014, pp. 675–678, doi: 10.1145/2647868.2654889.