Grenze International Journal of Engineering and Technology, June Issue

A Framework to Recognize Multiple Heterogeneous Face-Masks in Real Time using FusionNet

Bibhas Das[1], Souraneel Mandal[2], Sajib Saha[3] and Dipak kumar Ghosh[4]
[1-3]Adamas University Department of Computer Science and Engineering, Kolkata, India
Email: bd.bibhas@gmail.com, souraneel97@gmail.com, sajibsaha859@gmail.com
[4]Department of Electronics and Communication Engineering National Institute of Technology, Raipur, India
Email: dipakkumar05.ghosh@gmail.com

***Abstract*— COVID-19 pandemic has had a dramatic impact on our daily lives, disrupting global trade and transportation. Protecting one's face with a mask has become the new normal. In the present new normal situation and in near future also, customers must wear masks properly in many public services. Therefore, face mask identification has become a necessary and challenging stuff in the present new normal scenario. In this paper, a computer vision-based method using proposed convolution neural network (CNN), named as FusionNet is introduced to correctly detect the face mask. The proposed FusionNet architecture is constructed by combining of VGG16Net with MobileNetV2. This approach accurately recognizes the face mask from a video sequence and determines whether a person wearing a mask or not. The method is evaluated on the dataset which consist of total 400 images (200 images with mask and 200 images without mask) captured from different social gatherings. The proposed method can also be used to detect multiple faces with or without mask in a single video frame and achieves 99.75% accuracy which is better compared to individual performance of using standard VGG16Net and MobileNetV2.**

***Index Terms*— Covid-19, Face mask detection, FusionNet, VGG16Net, MobileNetV2.**

I. INTRODUCTION

Confront covers are basic needs for anticipating and controlling the spread of a few respiratory viral maladies, such as COVID-19. Confront veils can be utilized to secure sound people or to avoid undesirable people from contaminating them. Be that as it may, suitably employing a confront veil is pivotal to constraining defilement perils. In congested ranges such as a railway station, office, or school, the World Health Organization (WHO) [14, 15] suggests wearing a confront cover. A computer vision framework is competent of identifying whether or not an individual is wearing a confront cover or not. The quick development of computer vision has drawn expanded consideration around the world to cure from the infection of Covid-19 which develops human-computer interaction and also improve open well beings of administrations. Different countries are encountering a critical wellbeing catastrophe to reduce the (Covid-19) fast's spread.

Physically observing people in open areas and recognizing the confront veil within the video is very challenging task. Typically for the most part of human faces, veil itself acts as an obstacle to the confront. Since there are no confront signs within the cover locale it is very difficult to calculate distinguish proof calculations. As a result, the programmed confront veil acknowledgement innovation helps specialists in order to stop contamination. This

Grenze ID: 01.GIJET.8.2.525

work proposes a Computer Vision based method using FusionNet to recognize confront covers in live spilling. The proposed framework utilizes a live video bolster for various faces and is able to show efficiently whether somebody is wearing a veil or not with acceptable precision.
The rest of the paper is organized as: Section II presents an outline of the related works; Section III describes the proposed methodology; Result and Discussion mentioned in Section IV and Section V conclude the work.

II. Literature Survey

In **2001, Viola and Jones** [1] had expressed traditional approaches for extracting features which were employed in the construction of machine learning classifiers to detect objects. The most prominent one is the Viola-Jones detector, which detects faces by combining Haar characteristics with an integral picture method.
In **2008, Dalal et al.** [2] had proposed on a useful feature extractor called histogram of oriented gradients (HOG) is a feature extraction technique that computes the directions and magnitudes of oriented gradients over image cells for human detection, and scale-invariant feature transform (SIFT) is very useful for image matching and object recognition.
In **M. Inamdar et al. (2018)** [3-6] had expressed their work on a computer vision approach which deals with the intrinsic challenges of pattern learning and object identification. Object recognition includes picture categorization as well as detecting the objects within the image frame. An efficient object recognition technique through surveillance equipment can be implemented to recognize the mask over the face in public places. The pipeline of object recognition begins with the generation of region suggestions, which is followed by the categorization of each proposal into a related class. Single and double stage detectors with a genetic technique approach has been discussed by the authors as a recent advancements in region proposal to increase the detection performance and pre-trained the models.
Girshick et al. (2015) [7] had proposed a two-stage detector, in place of a single-stage detector, to predict and categorize area suggestions. The possible detection have been forecasted through two stages: picture suggestion prediction and application of classifier for finding the location. Various two-stage region proposal models have already been developed by academics. The region based convolutional neural network (R-CNN) described in 2014 is one of the earliest large-scale applications of CNN for object detection and localization application.
Furthermore, **Fan et al.** introduced a hybrid fast R-CNN and Region Proposal Network (RPN) **in 2016** [8] **& 2017** [9]. It allows for almost cost-free region proposals by gradually integrating separate blocks of the object detection system (e.g., bounding box regression, feature extraction and proposal detection) in a single step. Although this integration results in a breakthrough for the Fast R-CNN performance bottleneck, there is a computation duplication in the subsequent detection step.
Redmon et al. [10] had proposed a popular single-stage technique with amazing detection speed by displaying real-time predictions. However, it suffered from low localization accuracy when tiny objects were considered. The YOLO network splits an image into GxG grids, and each grid gives N predictions for bounding boxes. During the prediction, each bounding box can only contain one class, limiting the network's ability to discover smaller items. In addition, the YOLO network was upgraded to YOLOv2, which incorporated batch normalization, a high-resolution classifier, and anchor boxes. Furthermore, YOLOv3 is based on YOLOv2 which includes a new network for feature extraction and multi-sale prediction.
In 2018 **Sandler et al.** proposed MobileNetV2 Architecture [11], in contrast to standard residual models, which use extended representations in the input and output, the MobileNetV2 architecture is built on an inverted residual structure, where the input and output of the residual block are thin bottleneck layers. In the intermediate expansion layer, MobileNetV2 uses lightweight depth wise convolutions to filter features. In order to retain representational power, it is also discovered that non-linearities in the thin layers must be removed. In contrast to standard residual models, which use extended representations in the input and output, the MobileNetV2 architecture is built on an inverted residual structure, where the input and output of the residual block are thin bottleneck layers. In the intermediate expansion layer, MobileNetV2 uses lightweight depth wise convolutions to filter features. In order to retain representational power, the authors also discovered that non-linearities in the thin layers must be removed.

III. Proposed Methodology

This work proposes a new CNN architecture called FusionNet shown in Fig. 1. The proposed FusionNet is constructed by combining of VGG16Net [16] and MobileNetV2 [11] for face mask detection. In FusionNet, the final flattened layer of VGG16Net and MobileNetV2 are fused by concatenated and brought a new flatten layer.

Finally, the classification score is obtained by connecting the final flatten layer with the output layer through fully connected approach.

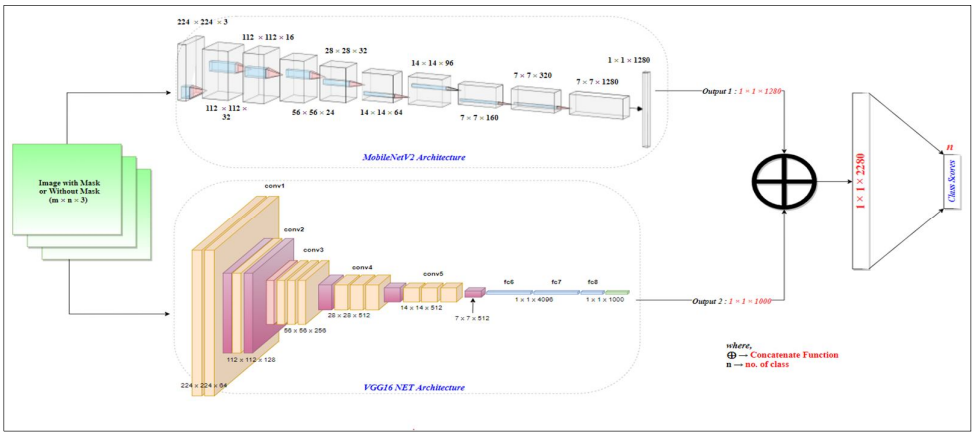

Figure 1: The architecture of Proposed FusionNet

The proposed face mask detection method using FusionNet consist of three major steps, such as ***Training, Implement*** *and* ***Deployment***. The details including sub-steps are shown in Fig. 2.

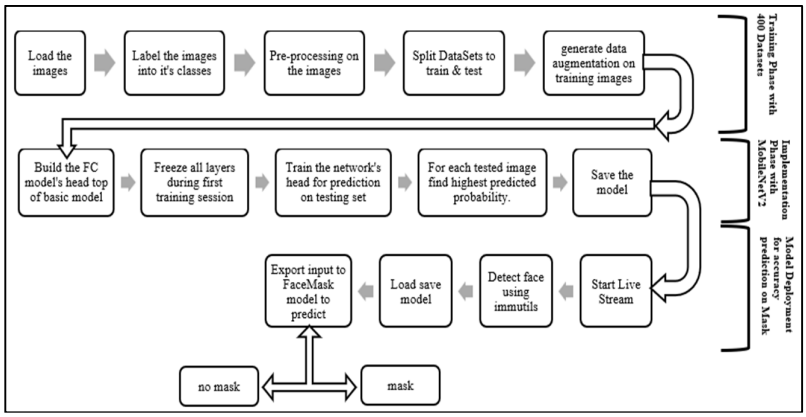

Figure 2**:** Flow chart of face mask detection using proposed FusionNet

The flow chart in fig. 2 is described as below:

Training:

Loading our face mask dataset from the disc and training the proposed CNN model. The details of training are listed below.

Step 1: Import the appropriate packages.
Step 2: Set the starting learning rate, number of iterations to train for, and batch size.
Step 3: Get the list of photos in our dataset directory, then initialize the list of data (i.e., images), then classify the images.
Step 4: Perform one-time encoding on the labels.
Step 5: Create the training image generator for data augmentation.
Step 6: Load the proposed network, making sure the head fully connected (FC) layer settings are turned off.
Step 7: Build the model's head, which will be put on top of the basic model.
Step 8: Place the head FC model on top of the base model (this will become the actual model which is trained).
Step 9: Cycle over all of the layers in the basic model and freeze them so that they are not modified during the first training session to compile the model.
Step 10: Train the network's head to make predictions on the testing set.

Step 11: For each image in the testing set, we must get the index of the label with the highest predicted probability.
Step 12: Provide a well-prepared categorization report.
Step 13: Save the model to disc.

Implementation:

Once the face mask detector has been trained, loading it, conducting face detection, and categorizing each face as with mask or without mask.

Step 1: Import the appropriate packages
Step 2: Obtain the dimensions of the frame and then create a blob from them.
Step 3: Run the blob through the network to get the face detections.
Step 4: Initialize the collection of faces, their related locations, and the list of predictions from the face mask network.
Step 5: Compute the probability and minimize false positives by ensuring that the probability is greater than the detection's lowest confidence level.
- Determine the (x, y)-coordinates of the bounding box of the item by ensuring that the bounding boxes are inside the
- frame dimensions. Pre-process the face ROI by converting it from BGR to RGB channel ordering, scaling it to 224x224.
- Add the face and bounding boxes to their respective lists.

Step 6: Make predictions only if at least one face is spotted.
Step 7: Rather than making one-by-one predictions, this work done batch predictions on all faces at the same time, producing a 2-tuple of face locations and their associated positions for quicker inference.

Deployment:

Step 1: Initialize the video stream and load our serialized face and mask detector model from disc.
Step 2: Iterate through the video stream frames.
Step 3: Take a frame from the threaded video stream and resize it to a user-defined maximum width of pixels.
Step 4: Look for faces in the picture and determine whether or not they are wearing a mask.
Step 5: Iterate over the observed face locations and their associated locations, unpacking the bounding box and making predictions. Choose the color and class label for the enclosing box and text.
Step 6: Display the label and bounding box rectangle on the output frame.

IV. Result and Discussion

To evaluate the performance of the proposed method, the experiments are carrying out a self-created database. The details of developed database is given bellow. The database contains total 400 images (200 images with mask and 200 images without mask) captured from different social gatherings. Each image resolution is 96 dpi having both RGB and B&W colors. Few sample images of the created database are shown in Fig. 3.

Figure 3: The sample images of developed database "with mask" and "without mask"

The dataset is divided into training and testing set in the ratio of 80:20 respectively. Train dataset is used to train the model where test dataset is used to evaluate the performance of the proposed model with FusionNet. The performance of the mask detection model is measured on the basis of its ability to detect cases and classify samples with mask and without mask.
The performances of the proposed method are evaluated in terms of precision, recall, f1-score and accuracy which are defined as follows.

The ***precision*** [11, 12] is the proportion tp / (tp + fp) where tp is the number of genuine positives and fp the number of wrong positives. The exactness is instinctively the capacity of the classifier not to name as positive a test that's negative.

$$\text{precision} = \frac{\text{TruePositives}}{\text{TruePositives} + \text{FalsePositives}}$$

The ***recall*** [11, 12] is the proportion tp / (tp + fn) where tp is the number of genuine positives and fn the number of untrue negatives. The review is instinctively the capacity of the classifier to discover all the positive tests.

$$\text{recall} = \frac{\text{TruePositives}}{\text{Predicted Results}}$$

The ***f1-score*** [11, 12] can be translated as a consonant cruel of the exactness and review, where an F1 score comes to its best esteem at 1 and most noticeably awful score at 0. The relative commitment of accuracy and review to the F1 score are break even with.

$$F1 - \text{score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

The ***accuracy*** [11, 12] Overall, how frequently is the classifier redress.

$$\text{Accuracy} = \frac{\text{TruePositives} + \text{TrueNegative}}{\text{Total}}$$

The performance accuracy with calculated loss of the proposed model using FusionNet during training and testing for 50 iterations are shown in **Table 1**. It is observed that after the completion of 50 iterations the proposed model using FusionNet provides training and testing accuracy of **99.74%** and **99.48%** resprecatively whereas training and testing loss of **1.02%** and **1.66%** respectively**.** A comparative study in terms of precision, recall and f1-score of the proposed model using FusionNet with VGG16Net and MobileNetV2 is shown in **Table 2.** The results indicate that proposed model achieves precision, recall and f1-score **0.99, 1.00** and **0.99** respectively with mask and **1.00, 0.99** and **0.99** respectively without mask images which are better compared to individual performance of MobileNetV2 and VGGNet16.

TABLE I. PERFORMANCE ANALYSIS OF TRAINING LOSS, TRAINING ACCURACY, VALIDATION LOSS AND VALIDATION ACCURACY

Iterations	Loss	Accuracy	Value Loss	Value Accuracy
1/50	0.4105	0.8521	0.1423	0.9844
10/50	0.0408	0.9862	0.0266	0.9948
20/50	0.0243	0.9928	0.0202	0.9935
30/50	0.0161	0.9924	0.0173	0.9948
40/50	0.0142	0.9954	0.0208	0.9935
50/50	**0.0102**	**0.9974**	**0.0166**	**0.9948**

TABLE II. COMPARATIVE PERFORMANCES OF THE MODEL WITH PROPOSED FUSIONNET, MOBILENETV2 AND VGGNET16.

Network used	Cases	precision	recall	f1-score
VGGNet16 [16]	with mask	0.98	0.96	0.95
	without mask	0.975	0.97	0.956
MobileNetV2 [11]	with mask	0.97	0.96	0.97
	without mask	0.98	0.967	0.97
Proposed FusionNet	**with mask**	**0.99**	**1.00**	**0.99**
	without mask	**1.00**	**0.99**	**0.99**

The graphical representation of training and testing accuracy/precision and loss/erosion of the proposed model using FusionNet is shown in **Fig. 4** and it shows that after 20 iterations training and testing accuracy/precision and loss/erosion are almost constant.

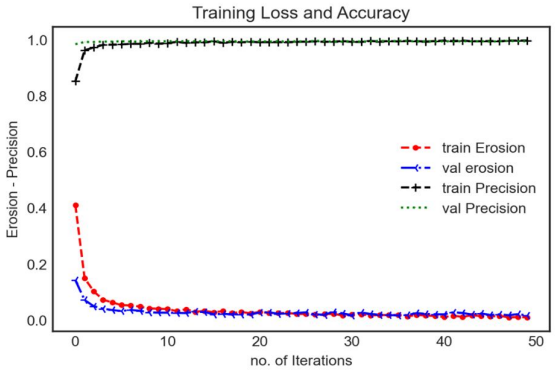

Figure 4: The training and testing accuracy/precision and loss/erosion of proposed model using FusionNet

To examine the mask detection capability and prediction accuracy of the proposed model, the experiments are carried out on ***live stream videos*** which includes single face image, double face image and multiple face image. The experimental outcomes are shown in **Fig. 5, Fig. 6 and Fig. 7** respectively.

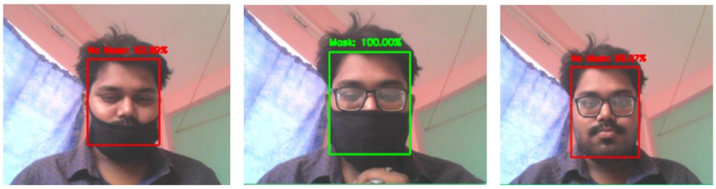

Figure 5: Sample mask detection outcome for single face image

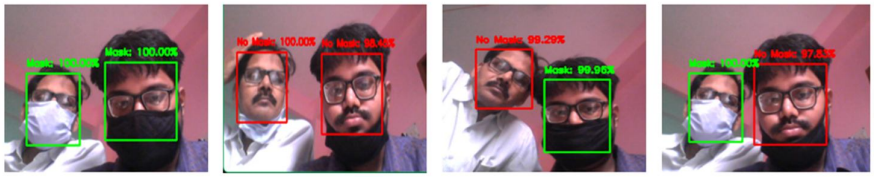

Figure 6: Sample mask detection outcome for double face image

Figure 7: Sample mask detection outcome for multiple face image

It is clearly visible from **Fig. 5**, **Fig. 6** and **Fig. 7** that our algorithm is capable to **identify mask or without mask** from **live stream videos** which includes single face image, double face image as well as multiple face image.

V. CONCLUSIONS

This work proposes a vision-based face mask detection model using proposed FusionNet architecture to identify whether a person wearing a mask or not. The performance of the proposed method is evaluated on self-created database which consist of total 400 images (200 images with mask and 200 images without mask) captured from different social gatherings. The proposed model achieves training and testing accuracy of **99.74%** and **99.48%** respectively. The experimental result also shows that mask detection model using proposed FusionNet gives better performance compared to individual performance of well-known MobileNetV2 and VGGNet16 architectures. The mask detection capability and prediction accuracy of the proposed model also evaluated on live stream videos and our model also successfully detect the mask from single face image as well as multiple face image in live stream videos.

REFERENCES

[1] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[2] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, pp. 886-893 vol. 1, doi: 10.1109/CVPR.2005.177.

[3] P. Felzenszwalb, D. McAllester and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1-8, doi: 10.1109/CVPR.2008.4587597.

[4] N. El Makhfi, R. Benslimane, "Feature extraction in segmented words for semi-automatic transcription of handwritten Arabic documents", J. of Theoretical and Applied Information Technology, Vol. 70. No. 1, pp. 68–75, 2014.

[5] M. Inamdar, N. Mehendale, "Real-Time Face Mask Identification Using Facemasknet Deep Learning Network", SSRN Electron. J. (july 29, 2020).

[6] S. Qiao, C. Liu, W. Shen and A. Yuille, "Few-Shot Image Recognition by Predicting Parameters from Activations," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7229-7238, doi: 10.1109/CVPR.2018.00755.

[7] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 1, pp. 142-158, 1 Jan. 2016, doi: 10.1109/TPAMI.2015.2437384.

[8] Cai, Z., Fan, Q., Feris, R.S., Vasconcelos, N. (2016). "A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection." In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds) Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science (), vol 9908. Springer, Cham. https://doi.org/10.1007/978-3-319-46493-0_22.

[9] C.Y. Fu, W. Liu, A.Ranga, A. Tyagi, AC Berg, "Dssd: Deconvolutional single shot detector", arXiv: 1701.06659 (2017).

[10] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[11] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. -C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510-4520, doi: 10.1109/CVPR.2018.00474.

[12] https://scikitlearn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html

[13] https://scikitlearn.org/stable/modules/generated/sklearn.metrics.f1_score.html?highlight=f1#sklearn.metrics.f1_score

[14] https://www.who.int/emergencies/diseases/novelcoronavirus-2019/global-research-on-novel-coronavirus-2019-ncov

[15] https://www.euro.who.int/en/health-topics/health emergencies/coronavirus-covid-19/novel-coronavirus-2019-nc

[16] Cimpoi, M., Maji, S., Vedaldi, A.: Deep filter banks for texture recognition and segmentation. In: CVPR, pp. 3828–3836 (2015)