ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/358367000

A Solution to Covid-19: Detection and Recognition of Faces with Mask

Conference Paper · February 2022

CITATIONS
0

READS
54

4 authors, including:

Roopashree Shailendra
RV Institute of Technology and Mangement

27 PUBLICATIONS **138** CITATIONS

SEE PROFILE

All content following this page was uploaded by Roopashree Shailendra on 05 February 2022.

The user has requested enhancement of the downloaded file.

2021 International Conference on Intelligent Technologies (CONIT)
Karnataka, India. June 25-27, 2021

A Solution to Covid-19: Detection and Recognition of Faces with Mask

Nithyashree V
Department of Computer Science and Engineering
Dayananda Sagar University
Bengaluru, India
nithyashree107@gmail.com

Roopashree S
Department of Computer Science and Engineering
Dayananda Sagar University
Bengaluru, India
roopashaily@gmail.com

Aparna Duvvuri
Department of Computer Science and Engineering
Dayananda Sagar University
Bengaluru,India
aparnaduvvuri2001@gmail.com

Vanishree L
Department of Computer Science and Engineering
Dayananda Sagar University
Bengaluru, India
mitra.vrinda07@gmail.com

Disha Anand Madival
Department of Computer Science and Engineering
Dayananda Sagar University
Bengaluru,India
dishaanand2016@gmail.com

Vidyashree G
Department of Computer Science and Engineering
Dayananda Sagar University
Bengaluru, India
vidyakg262@gmail.com

2021 International Conference on Intelligent Technologies (CONIT) | 978-1-7281-8583-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CONIT51480.2021.9498426

Abstract— **In this COVID-19 crisis, wearing masks is necessary and no longer an option to the general public. To follow the strict directives given by the government, the businesses have to implement a cost-effective approach to ensure that all its employers wear a face mask and help to control the spread of coronavirus. The proposed solution consists of an automatic face mask detection system that eliminates the need of an employee at the entrance. The working model detects a face mask in every person by analysing each frame of the video and alert through security mail when the mask is not detected. Our proposed system is designed using the Convolution Neural Network (CNN) model for mask detection and image subtraction technique to recognise the faces with mask. The scope of the project pertains to avoid the entry of unauthorized people into an organization by recognizing the face despite the presence of a mask. The model shows an accuracy of 99.82% on a custom dataset. It is an effective protection step to impede the transmission of the novel coronavirus.**

Keywords— COVID-19, Convolution Neural Network, Deep Learning, Image Subtraction, Masked Face Recognition, OpenCV, SMTP, SSL.

I. Introduction

Covid-19, a pandemic virus that hit human health recently [1]. It is declared as global pandemic by the World Health Organization (WHO). The statistics show that millions of cases were affected due to Covid-19 across 188 countries. In this paper, an approach for detection of face mask and recognition of the face with mask is proposed. The main objective is to help and monitor the spread of the virus focusing on the safety of the organization against the entry of intruders. A cost-effective solution that can be used efficiently by stores and businesses ensuring all its occupants wear facemask can be built using deep learning techniques. Computer Vision commonly known as CV, the technology involved in helping computers "see" and learn from images (photos) and videos. CV is a multidisciplinary field and its parent domain is artificial intelligence and machine learning, which utilizes special techniques of general learning algorithms [10]. The aim of computer vision is to understand the digital image contents by extracting more information from it, such as a text description, an object, a three-dimensional model, and many more [10]. It enables to recreate human vision. The required data can be gathered from views of multiple cameras, video sequences or multidimensional data [10].

Computer Vision applies numerous theories and models to develop many vision systems. The open source library named as Open Source Computer Vision Library (OpenCV) to solve computer vision and machine learning problems.

Deep learning models have proved to provide robust and effective solutions from videos and images as input using the convolution neural network which consists of many layers. The computer learns complicated concepts effectively from the hierarchy of concepts by building it from the simple ones. A graph plotted show how the concepts are built deep with many layers. The frameworks used for face mask detection are OpenCV, Keras, TensorFlow and Convolutional Neural Network (CNN). Image comparison techniques using OpenCV and python are used for face recognition. Image subtraction, similar to [5] was used for face recognition unlike the conventional technique which is widely used for face recognition [2]. It can also be extended to adaptive background subtraction [6] in-order to focus on the employee (our foreground alone). It can also be achieved through image segmentation, to separate the employee (the required foreground) from the background, as mentioned in [4], for even better performance and achieve background independency. SMTP (Simple Mail Transfer Protocol) and SSL (Secure Socket Layer) mail requests are used to alert the security.

Deep learning methods [10] focuses on learning features from higher levels of the hierarchy formed by the formation of low-level features. When features are automatically learnt through various levels of abstraction, then the system learns complex functions which map the input to output instantly from data and it is also partially dependent on human-crafted features. Deep learning techniques utilize the hidden structure in the input distribution to find good representations at multiple levels. The high-level features are defined in terms of low-level features.

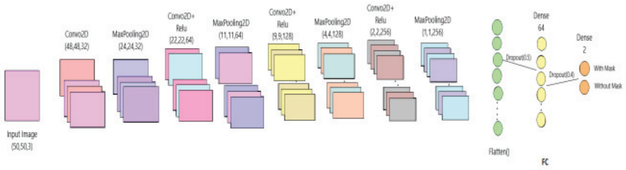

Fig. 1. Detailed CNN architecture for masked classification.

In deep learning, CNN architecture is used to train and test its models. For example, image classification using CNN accepts image as an input, processes and classifies the image to certain categories. Here the sequential model has 4 convolutional neural networks which is connected to a dense layer consisting of 64 neurons. Finally, we have 2 neurons which checks for the mask. The role of CNN is to down sample the images to a form easier to process without losing critical features essential for prediction [9]. CNN is widely used for image classification task[3]. It is thus chosen to classify masked and unmasked individuals. CNNs have four layers namely - Convolution, Pooling, ReLu and Fully Connected (flattening) layer. The convolution layer extracts the high-level features like edges from every frame (image) of the input video. In this layer, the kernel or filter performs convolution on the first part. The pooling layer [9] is needed to extract the dominant features which are independent of position and rotation of images to train the model effectively. Maxpooling and average pooling are the two types of pooling. In our case, max pooling is used to return the maximum value from the portion of the image covered by the kernel.

The SoftMax activation function is used to build the model using CNN along with ReLu activation function. The SoftMax transforms all inputs in the range of 0 to 1 to interpret the values in terms of probabilities. The ReLu function converts all negative inputs to zero. The Adam optimizer is used to minimize errors which uses an adaptive learning rate and a stochastic optimization algorithm. Finally, a dropout layer is added to avoid overfitting. The dropout layer sets inputs to zero randomly with a frequency of the rate at each step during training [9]. The flattening layer is used for image classification-with mask or without mask.

TensorFlow is an open source and free platform to solve many machine learning problems. It assists developers to build ML-powered applications easily. TensorFlow was developed by Google Brain of Google's Machine Intelligence Research organisation to conduct research on neural networks and machine learning.TensorFlow offers many API's for C++ and python and also offers great advantage of abstraction for solving machine learning problems. Using TensorFlow, the developers focus on the logic of the application rather than dealing with the implementation of its algorithm. TensorFlow being an open-source software library provides python interface for artificial neural networks.

Keras, written in Python, supports multiple back end neural network computation engines, is one of the leading APIs for high-level neural networks. It is a Python library that runs over TensorFlow for deep learning.

Google Collaboratory notebook (Google Colab), a Google Research product, is a free online cloud-based Jupyter notebook environment. It allows to train the models on GPUs, CPUs and TPUs related to machine learning and deep learning. Many popular libraries are pre-installed on Google Colab namely pandas, numpy, scikit-learn and many more. Anybody can write and execute the python programs related to machine learning easily using Google Colab. Thus, Colab proves to be more beneficial, popular and powerful over jupyter notebook as it does not require any setup. It provides access to computing resources such as GPUs and ability to choose various types of runtimes.

II. Related Works

There are many works related to the recognition of faces. Out of which, the recognition of non-frontal faces [7] showed good results. But, in the work, pixel accuracy decreases as the number of people increases to detect the faces in order to fit the position of people to the size of the camera window. A study [8] based on photo-photo identification in facial recognition systems which focused on the implementation of deep learning techniques for recognizing the faces using forensic sketches to facial photograph matching. However, limitations in handling pose variations, occlusions and video data were observed.

A project [10] illustrates the implementation of various technologies such as OpenCV, keras, tensor flow, and CNN to identify the face masks with a decent accuracy. The work proposes on further optimization and review of major efforts and advances in the field of face recognition, focusing on all types of variations in facial images. However, to extract more robust features, large data set is essential and hence to complete the process of monitoring large groups of people proves to be a difficult task.

A novel face mask detector is a vital contribution to the public healthcare domain. We have developed face mask detection model using an online dataset to build the CNN model so that it detects the presence of a face mask in each and every frame of the input video. Our work also detects the multiple facial masks in a single video which consists of 40 frames approximately. CNNs are very convenient and hence are widely used in many image classification tasks.

III. Methodology

We propose a two-phase COVID-19 face mask detection system, which performs the detection of face mask in phase-1 and recognition of masked face (employee) in phase-2. Implementation details to detect the face mask using computer vision and deep learning techniques are revealed using the custom-built data set. We have used two datasets namely, Mask detection dataset shown in Fig.2 and organization dataset shown in Fig.3. The mask detection dataset has images of masked and unmasked people. So, this dataset is used in face mask detection phase. The organization dataset has images of the masked employees used in employee face recognition phase.

The proposed two-phase face mask detection system, can detect the presence or absence of face mask by training the input image in an appropriate convolutional neural network with both fully connected layer as shown in Fig. 4 and SVM final layers shown in Fig. 5 to get accurate results. The experimental results show that the accuracy of CNN classifier is higher than SVM.

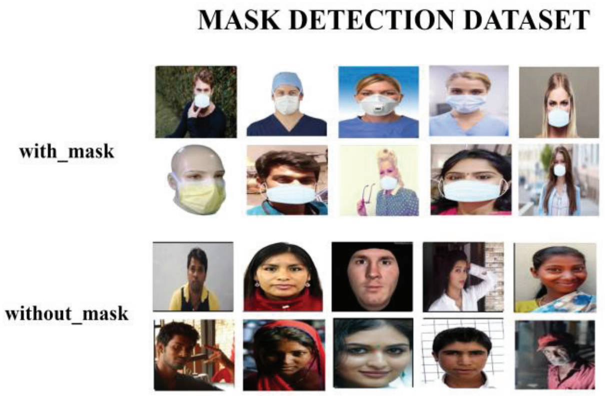

Fig. 2. A sample dataset for mask detection

Fig. 3. Organization dataset

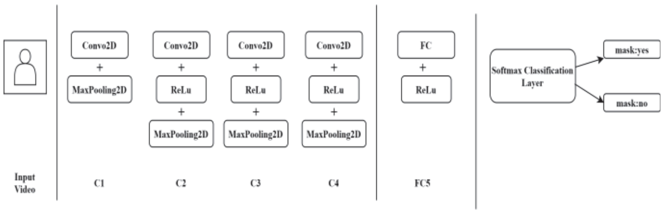

Fig. 4. Proposed CNN model

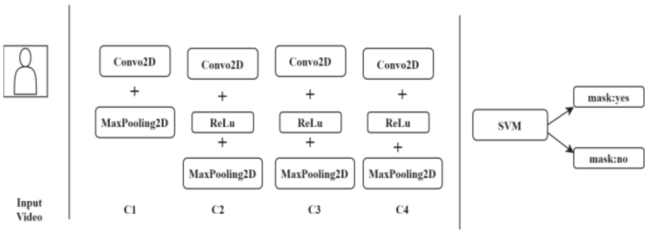

Fig. 5. Proposed SVM model

The final result of our proposed system is based on the CNN classification.

The proposed model requires an RGB image as an input of any arbitrary size. The basic function of the model is feature extraction and prediction of face-mask. In the second phase, face recognition of the image is carried out to determine if the masked individual belongs to a particular organizational premise.

A. *Proposed Workflow*

We propose a method of detecting face mask (MASK: YES, or MASK: NO) from the input video. The input video considered has approximately 40 frames. Each of these frames is resized to 50x50x3 and fed to the FCN network and also to the SVM final layer for feature extraction and prediction regarding the face-mask. However, the final decision is based on two cases such as (1. If the mask is not worn in the query image, the output will be MASK:NO and a pandemic alert mail is sent to the security using SMTP protocol. 2. If the mask is detected in the input image, the output produced will be MASK: YES from the application and further forwarded to the employee face recognition phase. In this phase, the last frame of the video from phase-1 will be the input image which is, later compared to all other images in the organization dataset. If match found, the person whose record is being compared proves to be an employee of the organization, otherwise the person does not belong to the organization and is considered an outsider/intruder. As a result, an organization security alert mail is sent to the security along with the intruder's image.

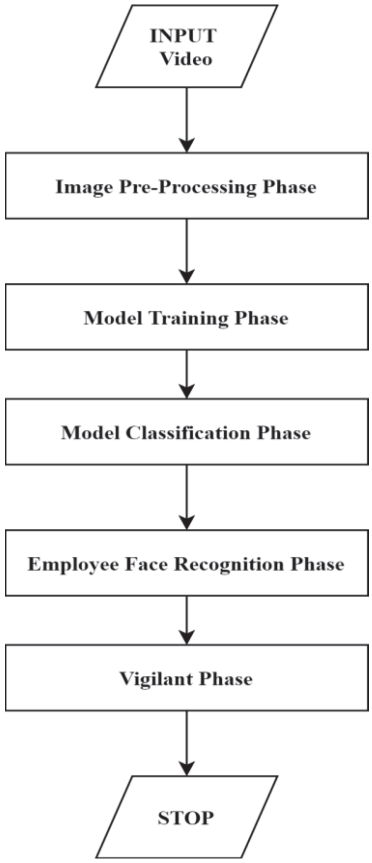

Fig. 6. Proposed System Flowchart

1) *Image Pre-processing:*

Initially the system visualizes all the images in the dataset i.e. both masked and unmasked images, then data augmentation is done. Further for reading the image, cv2.imread() is used. Later the RGB image is converted into grayscale image and resized to 50X50 dimension and the image is finally converted into 3D form.

2) *Model Training:*

In training phase, the data is split into training and testing set with the ratio of 80:20. The train set constitutes of images on which the proposed CNN model is trained and a separate test set used for testing. The CNN model is built using Keras library with 50 epochs. At this point the model is saved using model.save ().

3) *Model Classification -Face Mask detection:*

Then comes classification (face mask detection) where the CNN model is loaded using keras.models.load. Later pre-processing of the video captured at the front gate of the organization is performed. Now labelling the result is done using tf.mat.argmax for two probabilities: 0 for image with no mask and 1 for image with mask. At this instant, the result is displayed as follows: “Mask:Yes” if mask is detected else “Mask:No” if not detected i.e. mask not worn by an individual then a “Pandemic Alert” mail is sent to the security for further actions. For transferring the alert mail to the security, SMTP (Simple Mail Transfer Protocol) is used. The result is displayed on the top left-hand corner on each and every frame extracted from the input video of the employee captured at the gate.

4) Face Recognition:

If the mask is detected, the image (the last frame) enters the face recognition module (phase-2) for the purpose of authentication of the masked employee. This image is compared with the pre-existing masked images of all the ten employees of the organization using functions such as CountNonZero(), subtract() and split() for similarities. If both the images match then the individual is allowed to enter the organization. If no match is found, “Organizational Alert” mail is sent to the security using SMTP.

5) Face Mask Detection:

The module includes three simple steps - Data Pre-processing, training the CNN and finally detecting face masks. In data pre-processing, we walk through the folders, get all the images and store them in a list after which they are resized to 50x50 grayscale images. As grayscale images have only one channel, they are further converted to 3D images. We proceed to create 2 models such as CNN with softmax (classifier) and the other with CNN and SVM as the classifier. CNN with multiple layers using ReLu, softmax activation functions, Adam optimizer and max pooling technique in model 1. Finally, we add the dropout layer in-order to avoid over-fitting while training our model.

Here the sequential model has 4 convolutional neural network which is connected to a dense layer consisting of 64 neurons. Finally, we have 2 neurons and the SVM final layer which check for the mask. Finally using softmax we predict if the mask is worn (if one is returned) or not worn(if zero is returned) and the same is displayed on every frame of the input video.

6) Employee Face Recognition:

Unlike conventional recognition techniques here, image subtraction is used to achieve the same. If the face of the individual is masked, only then the system enters this module. The image is compared using subtract() method of OpenCV with every image of the organization list. Every image of the organization dataset is of size 50x50.The test image is also resized to 50x50.The subtract() method subtracts corresponding pixels of the test image with the masked employee images in the dataset.

Further, countnonzero() is used to count the number of non-zero pixels in the difference-if this is 0 for all blue, green and red, we declare that the test image is that of an employee and also display his/her id and name. If there is no match, we declare that the test image does not belong to the organization and hence display UNKNOWN along with this photo.

7) Vigilant phase:

The SMTP and SSL Python Libraries are used to send the two different alert mails to the organization’s security. The protocols, Secure Sockets Layer (SSL) and Transport Layer Security (TLS) are used to establish authenticated and encrypted links between networked computers. It is mainly used to send messages to other users using email addresses. The exchange of mail is provided between users on the same or different computers and a single message can be sent to many recipients. The messages can be text, video, voice or graphics and can be sent to networks outside the internet.

After the first phase (face mask detection), if an individual is unmasked then an alert mail with the subject line-”PANDEMIC ALERT!!!” is sent to the security with a warning message along with the image of the unmasked person at the gate. After masked face recognition is completed if there is no match found for the test image, then, an alert mail with the subject line-”ORGANIZATION ALERT” is sent to indicate that a masked person is at the gate who does not belong to the organization.

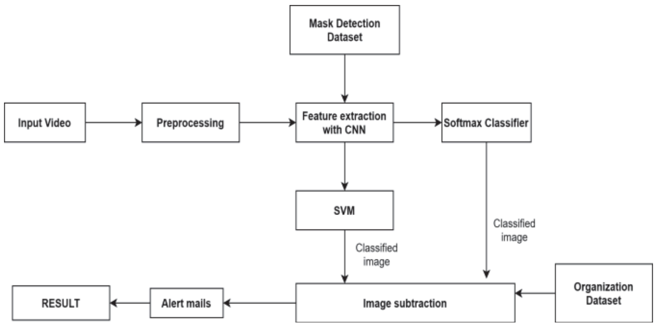

Fig. 7. Conceptual block schema.

B. Architecture

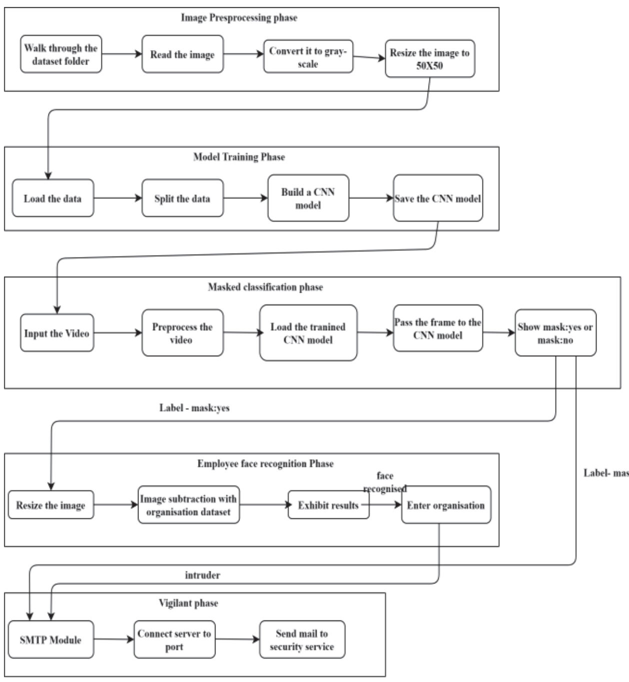

Fig. 8. Proposed architecture diagram.

Proposed algorithm is defined below:

Step-1: Image Pre-processing Phase: Walk through the Mask Detection dataset folders. Read input image. Convert to gray-scale and resize to 50x50.

Step-2: Model Training Phase: Load the pre-processed images. Split the images to the ratio of 80:20 (for training and testing). Build the CNN model from scratch for the training images. Model results with class labels-Mask: Yes or Mask: No. Save the model.

Step-3: Masked Classification Phase: Input the video. Pre-process each input frame by converting to grayscale and resize to 50x50. Load the trained CNN model. Pass each preprocessed frame as an image to the trained CNN model. Show the classified labels as Mask: Yes or Mask: No. If Mask: No then activate the Vigilant phase otherwise to Employee Face Recognition phase.

Step-4: Employee Face Recognition Phase: Input is the image from the previous classification phase. Perform image subtraction by comparing the input image with every employee image in the organization dataset folder. Exhibit the result label such as intruder and employee. The image subtraction will identify the person as either an employee or an intruder. If intruder, move to the next phase (Vigilant Phase) otherwise display the employee-id and name of the recognized employee.

Step-5: Vigilant Phase: Activate the SMTP module. Connect to the server through the port. Send mail to the security service along with the image. After unsuccessful recognition from Employee Face recognition phase, an organization alert mail is sent to the security. After unsuccessful detection of face mask from the Masked Classification phase, a pandemic alert mail is sent to the security.

IV. RESULT ANALYSIS

We have implemented both CNN-softmax and CNN-SVM models for face mask detection in order to compare the accuracies and to show the increased potential of deep learning techniques over the machine learning techniques. Fig. 9 and Fig. 10 reveals that the accuracy of CNN model is more than SVM model. Apart from the two graphs, the performance of the two models were also evaluated using precision, recall and f1-score. Clearly, from the classification reports of CNN and SVM models shown in TABLE I and TABLE II respectively, our CNN model performs better than the SVM model.

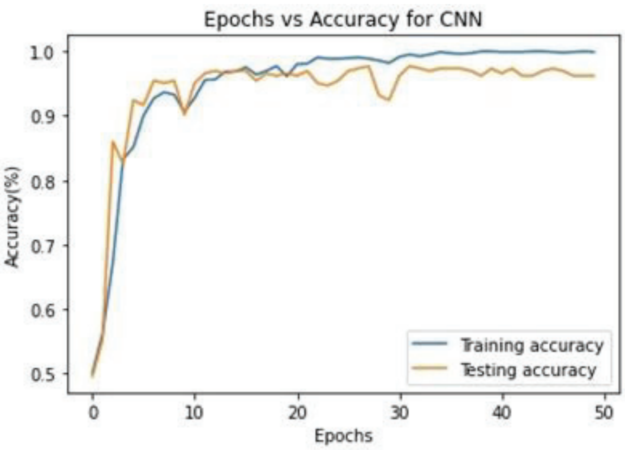

Fig. 9. Graph of variations in Accuracy with Epoch in CNN model.

In conclusion, we have implemented a deep learning model for face mask detection. The implemented SVM over the custom dataset achieve 92% testing accuracy. Though SVM is a very strong technique, achieving a high accuracy with SVM as final layer is still an exception. We achieved 97.32% training accuracy using SVM for the same dataset. Eventually, to achieve more accuracy than SVM, we implemented CNN with fully connected layer (softmax) as the final layer instead of SVM and achieved 99.82% training accuracy and 96.97%. testing accuracy.

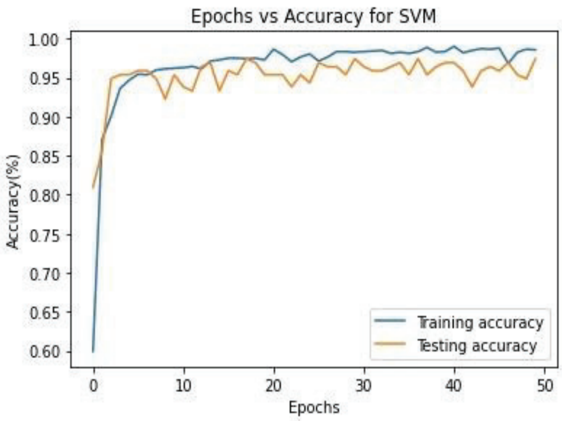

Fig. 10. Graph of variations in Accuracy with Epoch in SVM model.

TABLE I. PERFORMANCE EVALUATION OF CNN MODEL

	Precision	Recall	f1-score	Support
with_mask	0.95	1.00	0.97	156
without_mask	1.00	0.95	0.97	175
accuracy			0.97	331
macro_avg	0.97	0.97	0.97	331
weighted avg	0.97	0.97	0.97	331

TABLE II. PERFORMANCE EVALUATION OF SVM MODEL

	Precision	Recall	f1-score	Support
with_mask	0.91	0.99	0.95	156
without_mask	0.99	0.91	0.95	175
accuracy			0.95	331
macro_avg	0.95	0.95	0.95	331
weighted avg	0.95	0.95	0.95	331

V. CONCLUSION AND FUTURE WORK

The discussed research has been successful in developing a system that can detect face mask and hence recognize the masked individual. The proposed system potentially recognizes if the masked individual is an employee of the organization or not.

We have used two datasets namely Mask detection dataset and organization dataset, targeting a small organization using VSDC video editor software for the input videos of masked employees. Both the proposed CNN models are trained using 1320 images and tested on 331 images. The CNN model with softmax as the final layer showcased training accuracy of 99.62% and testing accuracy of 96.97%. The CNN model with SVM as final layer presented its training accuracy of 97.32% and testing accuracy of 95.88%.

Therefore, it is evident that the CNN model with softmax as final layer outperforms the other CNN model with SVM as final layer. The major challenge in face mask detection is the capability of detecting various types of colored masks with low error rate.

Proposed architecture can be deployed into a surveillance system of any organization to form a fully automated face mask detection and masked face recognition along with different alerts for unmasked individuals and intruders in the future. Besides, the masked employee’s attendance can also be marked by connecting the system to a database instead of using conventional biometric systems. To further improve the face mask detection model, actual images of masked faces to be gathered. Secondly, the face images that may “confuse” the classifier, such as cloth wrapped around faces,mouth, etc. to be gathered. The material of the mask such as cloth mask, surgical mask or 3 layered masks can be detected.

REFERENCES

[1] Liu, Xiaopeng, and Sisen Zhang. "COVID‐19: Face masks and human‐to‐human transmission." Influenza and Other Respiratory Viruses 14.4 (2020): 472-473.

[2] Coşkun, Musab, et al. "Face recognition based on convolutional neural network." 2017 International Conference on Modern Electrical and Energy Systems (MEES). IEEE, 2017.

[3] Chaganti, Sai Yeshwanth, et al. "Image Classification using SVM and CNN." 2020 International Conference on Computer Science, Engineering and Applications (ICCSEA). IEEE, 2020.

[4] Tarafdar, Arunabha, et al. "Image Segmentation Using Background Subtraction On Colored Images." 2019 International Conference on Opto-Electronics and Applied Optics (Optronix). IEEE, 2019.

[5] Lee, Hasup, HyungSeok Kim, and Jee-In Kim. "Background subtraction using background sets with image-and color-space reduction." IEEE Transactions on Multimedia 18.10 (2016): 2093-2103.

[6] Sakpal, Neha S., and Manoj Sabnis. "Adaptive background subtraction in images." 2018 International Conference On Advances in Communication and Computing Technology (ICACCT). IEEE, 2018.

[7] Meenpal, Toshanlal, Ashutosh Balakrishnan, and Amit Verma. "Facial mask detection using semantic segmentation." 2019 4th International Conference on Computing, Communications and Security (ICCCS). IEEE, 2019.

[8] Chokkadi, Sukhada, and Abhir Bhandary. "A Study on various state of the art of the Art Face Recognition System using Deep Learning Techniques." arXiv preprint arXiv:1911.08426 (2019).

[9] Jmour, Nadia, Sehla Zayen, and Afef Abdelkrim. "Convolutional neural networks for image classification." 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET). IEEE, 2018.

[10] Covid-19 facemask detection with deep learning and computer vision by Vinitha.V, Department of Master of Computer Application, AMC Engineering College Bangalore, Karnataka, India et al. (2020)

View publication stats