© July 2022 | IJIRT | Volume 9 Issue 2 | ISSN: 2349-6002

Real-Time Face Mask Detection Using Deep Learning

Roopa B[1], Pallavi V R[2], Raksha S[3], Rakshitha R[4]
[1,2,3,4] *ATMECE*

Abstract: **Pandemic on a world scale COVID-19 a virulent disease of hazardous sickness erupted in 19 countries throughout the globe. Wearing a mask can help reduce the spread of infection and also the transmission of infectious germs through the air. mask Detection System can detect whether people are wearing masks or not. The efficient feature selection, feature extraction and variety of false positive features are the key aspects. We require mask detection as a singular and public health service system during the planet pandemic COVID-19 outbreak supported this rationale. Mask and non-face mask images are accustomed train the model which helps in real time detection of the people wearing mask or not.**

INTRODUCTION

The world remains recovering from the widespread of COVID- 19, and a vaccine for its cure has yet to be discovered. to reduce the economic burden of the epidemic, several countries have allowed a restricted number of economic activities to restart once the number of latest cases has decreased. Covid-19 has fallen below a particular threshold. Concerns about worker safety have surfaced within the new post-Covid-19 climate as these countries cautiously recommence their economic activity. it's recommended that people wear masks and keep a distance of a minimum of 1 meter between them to limit the danger of infection.

Deep learning has gotten plenty of interest within the sector of object detection, and it's been accustomed produce a mask identification tool which can tell if someone is wearing a mask or not. this might be done by studying real-time streaming from the Camera and evaluating the categorization findings. A training data set is required for deep learning applications. As a result, detecting face masks has become a extremely important and difficult task. Face recognition without a mask is simpler, but face recognition for a typical face could also be efficient for feature extraction than a masked face. Many facial characteristics like the nose, lips, and chin, are all missing due to the covered face. According to the medical industry, wearing a mask minimizes the danger of being exposed to an infected person, whether or not they exhibit symptoms. But this has made the great deal in identifying people while they're wearing mask and also, it's become difficult for the system to detect the faces by extracting the common features.

LITERATURE SURVEY

1.MASKED FACE RECOGNITION FOR SECURE AUTHENTICATION

This paper explains about the problems associated with recognizing the faces of people who are wearing a mask through existing face recognition system with good accuracy. They address a methodology that uses the existing datasets by adding some features that enable the recognition of masked faces with less false-positive rate and overall high accuracy. This need not to be create a new dataset by taking pictures for authentication purpose. To create the masked faces dataset from face dataset they make use open-source tool called MaskedTheFace, this tool chooses the correct template of the mask based on the position of the face and estimate face tilt angle, finally wrap the mask according to estimated mask key positions. Then the dataset will be trained for effective facial recognition with reliable accuracy. By using this proposed system, they report an increase of ~38% in the true positive rate.

2.EFFICIENT MASKED FACE RECOGNITION METHOD DURING COVID-19 PANDMEIC

The proposed method is based on occlusion removal, this approach aims to remove the region other than the region covered by mask and discard them completely for feature extraction and classification process. Next a three pre-trained deep Convolutional Neural Networks, Namely Alex Net, ResNet-50, and VGG-16 are used to extract the features from the excluded part. After extracting the features, a Bag-of-features paradigm is used to map the features on the convolutional layers and Support Vector machine classifier is used to detect and get the slight representation of discarded region. Lastly Multilayer Perceptron is applied in classification process. The proposed method is also applied to the real- world-Masked-Face- Datasets and it performed well when compare to the state-of-the-art methods in terms of accuracy and time complexity.

3. FACIAL RECOGNITION SYSTEM FOR PEOPLE WITH AND WITHOUT FACE MASK IN TIMES OF THE COVID-19 PANDEMIC

This paper describes the system that is being developed to recognizing people, even if they are wearing a mask. This proposed system is design to identify the people faces with mask and without a mask. They have used two databases for the proposed system to work properly, the first one consists of large numbers of images of people faces with mask and without a mask dataset. The second is used in training and face recognition system. By taking this dataset we can provide the images and video as input to architecture used is Mobile Net. Their methodology is carried in three steps, The first step is to identify the location and dimension of one or more faces, the OpenCV face detection model is used that result in ROI and contains some data. In the second step Classification of faces will be done. Mobile Net V2 architecture is used as classifier this improves the performance resulting in a better accuracy. In the third step based on some feature facial recognition is carried out and set of observation are made. After evaluating the facial recognition model they report around 95% of accuracy for both the face dataset and masked face dataset.

METHODOLOGY

Face mask detection

Using the Convolution Neural Networks (CNN) algorithm in this paper, a deep learning approach is utilized to recognize face masks and classify them using Convolution Neural Networks (CNN). A CNN is a type of artificial neural network that is designed to understand pixel input and is mostly used for image recognition and analysis, with each layer applying a distinct set of filters. A large number of filters are combined to produce a final result, which is then delivered to the next layer in this neural network. The proposed framework is evaluated using the face mask detection technique and the TensorFlow software library, as illustrated in Fig 1.

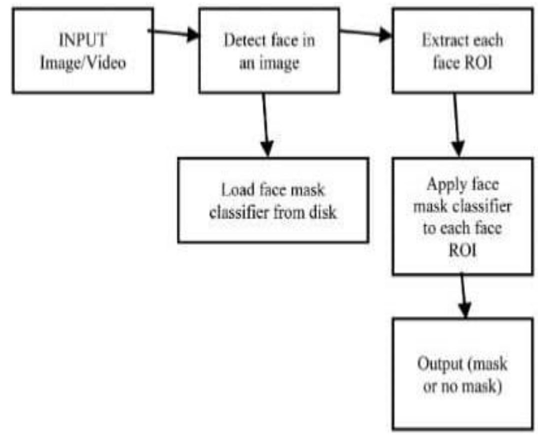


fig1. face mask detection

Keras and TensorFlow are used to train the Mask detection model. The following lists the steps that make up the algorithm.

STEP 1: COLLECTION OF DATASET
STEP 2: PRE-PROCESSING
STEP 3: SPLITTING
STEP 4: TRAINING
STEP 5: EVALUATION/TESTING

The network's entire dataset and component set are gathered from various categories in accordance with the aforementioned algorithm. The next stage is to train and test the dataset after it is complete. This test dataset is only intended to assess how well the network performs. Next training should be conducted so that the neural network may start recognizing various categories within the provided labels. Finally, it is important to assess the dataset and contrast it with the ground-truth labels. To find the bounding box region of a face in an image, begin with a picture of a person who is not wearing a face mask and then perform face detection. It can capture the face Region of Interest (ROI) after detecting where the face is in the image, and then use facial landmarks to recognize the position of the mouth, eyes, nose, and other features. To begin, an image of a mask is required, which will be automatically applied to the face using facial landmarks (especially the regions around the mouth and chin) to decide where the mask should be placed. The mask is then resized and twisted before being fitted to the face, and the procedure is repeated for each of the input photos.

Dataset Collection

Images from the internet were utilized to train and test the model. There are 1930 images without masks and 2166 images with masks in this collection, which has 4095 images total. They took common photographs of people's faces to construct this dataset, and then they added face masks to the images using a specially created computer vision Python application. The position of facial features including the eyes, nose, brows, mouth, and jawline may all be inferred instantaneously by the user thanks to facial landmarks. Following that, a dataset of faces wearing masks can be produced using facial landmarks. Start with a picture of a person who isn't wearing anything to determine the area of the face's bounding box. After figuring out where the face is in the image right now, it can capture the face Region of Interest (ROI) and use facial landmarks to identify the placement of the mouth, eyes, nose, and other features. First, a mask picture must be created. The mask will then be applied to the face automatically using landmarks on the face, notably the areas around the mouth and chin. The process is repeated for each of the input images, and then the mask is scaled and twisted before being fitted to the face. Resulting in the dataset given in Fig. 2 for an artificial face mask The face is captured, and the blob is built using images of people wearing and not wearing masks. This blob is sent over the network in order to detect faces from the extracted blob, and the trust (i.e., probability) is also connected with the extracted detection. The weak detection is filtered to ensure that the confidence (probability) is greater than the minimum degree of reliability, so that the face ROI (Region of Interest) is extracted and switched to RGB format from BGR format, and it is reformatted to 4 x 224, and then pre-processing is completed, and now the extracted face is sent via the mask detection model to detect whether the face sent is wearing a proper face mask or not. The weak detection is filtered to guarantee that the confidence (probability) is greater than the minimum degree of dependability. As a result, the bounding box and text are drawn, and the label includes probability. Finally, as a frame on the output screen, a white box known as a rectangle bounding box appears with a label.

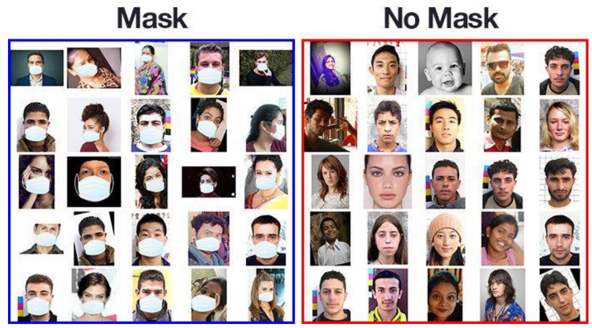


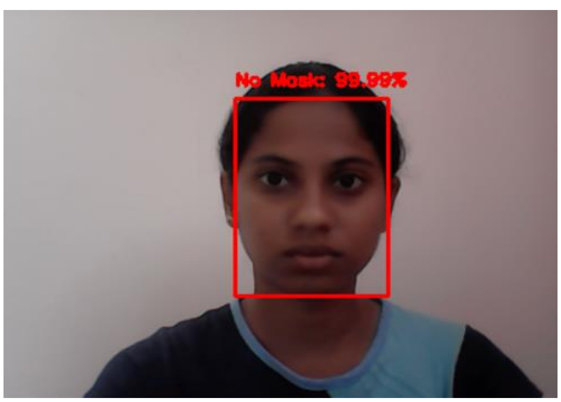
fig.2. people with and without mask

PSEUDO CODE FOR TRAIN AND TEST [4]

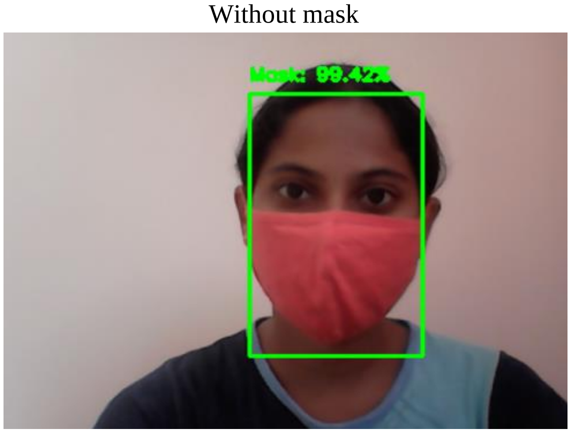
```
# initialize the initial learning rate, number of epochs
to coach for,
# and batch size
INIT_LR = 2e-4
EPOCHS = 20
BS = 32
DIRECTORY=r"C:\Users\RAKSHITHA\Desktop\F
inalProject\Face-Mask-Detection-master (2)\Face-
Mask-Detection-master"
# partition the info into training and testing splits
using 75% of
# the information for training and also the remaining
25% for testing
(trainX, testX, trainY, testY) = train_test_split(data,
labels, test_size=0.30, stratify=labels,
random_state=40)
# construct the training image generator
aug = ImageDataGenerator(
rotation range=20,
zoom range=0.15,
width_shift_range=0.2,
height_shift_range=0.2,
shear_range=0.15,
horizontal_flip=True,
fill_mode="nearest")
# load the MobileNetV2 network,
# left off
baseModel=MobileNetV2(weights="imagenet",incl
ude_top=False, input_tensor=Input(shape=(224,
224, 3)))
# train the pinnacle of the network
print("[INFO] training head...")
H = model.fit( aug.flow(trainX, trainY,
batch_size=BS),
steps_per_epoch=len(trainX) // BS,
validation_data=(testX, testY),
```

```
validation_steps=len(testX) // BS,
epochs=EPOCHS)
# make predictions on the testing set
print("[INFO] evaluating network...")
predIdxs=model.predict(testX, batch_size=BS)
# plot the training loss and accuracy
N = EPOCHS
plt.style.use("ggplot")
plt.figure()
plt.plot(np.arange(0, N), H.history["loss"],
label="train_loss")
plt.plot(np.arange(0,N), H.history["val_loss"],
label="val_loss")
plt.plot(np.arange(0,N), H.history["accuracy"],
label="train_acc")
plt.plot(np.arange(0,N), H.history["val_accuracy"],
label="val_acc")
plt.title("Training Loss and Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig(args["plot"])
```

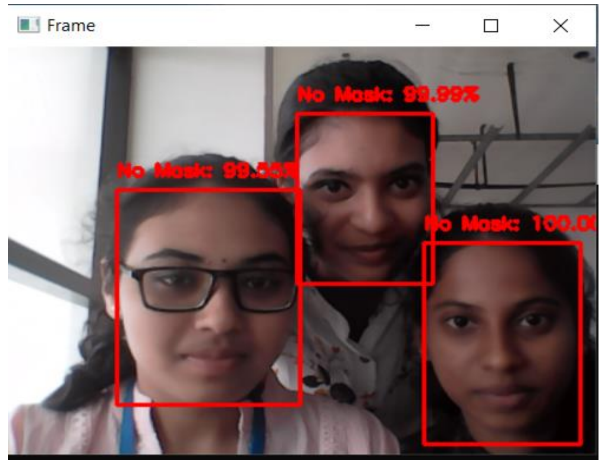
RESULTS



Without mask



With mask



Multiple faces

CONCLUSION

In real time, our proposed system can detect and recognize human faces. Face detection and recognition based on CNN model and Python libraries has shorter detection and recognition time and stronger robustness, which can reduce the miss rate and error rate. In a sophisticated environment, it can still guarantee a high-test rate, and the detection speed can meet the real-time requirement while achieving good effect. For detecting and recognizing human faces, the proposed CNN model performs better in terms of accuracy and prediction. The results show that the current technology for face detection and recognition is flawed and that this proposed work can replace it. As a result, the proposed method is successful. The proposed method works exceptionally well in biometrics and surveillance applications.

REFERENCE

[1] Roshan M Thomas, Motty Sabu, Tintu Samson, Shihana Mol, Tinu Thomas , “Real Time Face Mask Detection and Recognition using Python”,2021 International Journal of Engineering Research & Technology (IJERT)

[2] Hiten Goyal, Karanveer Sidana, Charanjeet Singh ,Abhilasha Jain, Swati Jindal,” A real time face mask detection system using convolutional neural network”,2022 Multimedia Tools and Applications (2022) Spinger.

[3] Aqeel Anwar, Arijit Raychowdhury, “Masked Face Recognition for Secure Authentication”, 2020

[4] Walid Hariri, “Efficient Masked Face Recognition Method during the COVID-19 Pandemic”, 2021
[5] Chandrikadeb7 ,”Face-Mask-Detection”, github