FACEMASK AND PHYSICAL DISTANCING DETECTION USING TRANSFER LEARNING TECHNIQUE

Kalaichelvi K[1], Tamilarasan R[2], Sureshbabu N[2], Ravi V[2], Santhosh Kumar S M[2]

[1]*Associate professor, Department of Electronics and Communication, V.S.B Engineering college, Karur 639004,Tamilnadu,India*

[2] *Department of Electronics and Communication, V.S.B Engineering college, Karur 639004,Tamilnadu,India*

---***---

Abstract—*The COVID-19 is a disease that is highly contagious and spreads from an infected person to another through droplets, when that person sneezes or coughs. The effective way to reduce the spreading of the virus is to use the face masks properly as enforced by the governments and also by following the social distancing. There are also survey reports which emphasize the effectiveness of masks and social distancing in the reduction of the spread of the COVID-19. Even Though the situation is well-handled by the government with people's support, the disease is not brought into control yet. So precautionary steps are needed to monitor people following the rules to bring the spread into control. For auto recognition of the people who are not wearing the face masks and violating social distancing, we can use the deep learning technology with the YOLOv3 algorithm, which is highly useful in object and image recognition and classification applications. Here we are implementing the same with both face masks and social distancing monitoring. For training the model datasets are collected from the internet which are freely available.*

Keywords : IoT, Deep learning, Image processing, YOLOv3.

I. INTRODUCTION

During the COVID-19 pandemic in many countries, wearing a mask is mandatory and indeed, it reduces the fatality rates. As long as the vaccine is not widely utilized and is not fully protective for every person, wearing a mask is crucial. As it is important to control the spread, the authorities need a way to monitor public places like subway stations and shopping centers. Therefore, there is a need to detect masked and unmasked faces. Because face detection is a vital part of this process, it requires a large amount of time and resources if done manually and increases the chance of making mistakes in detecting unmasked faces. Machine learning and computerized vision techniques can help automate this process.

It is difficult for the government to control the spreading without people following the instructions. Communication of this disease can only be reduced with the right cooperation from the public. Maintaining Physical distance, repeated hand sanitization, and face masks have all been shown to be effective in preventing the virus from spreading, but not everyone is following the rules. The spread of the coronavirus would be difficult for India to control. The most effective means to prevent transmission are face masks and hand sanitizers. This has had positive benefits in terms of minimizing disease transmission.

In the discipline of Computer Vision and Pattern Identification, facial recognition is a crucial component. This method has a number of disadvantages, including a high level of feature complexity and low detection accuracy. Face recognition approaches based on deep convolutional neural networks (CNN) have been widely developed in recent years to increase detection accuracy.

Several authors have employed predefined standard models as VGG-16, Resnet, and MobileNet, which need a lot of memory and computing time. An effort was made in this study to modify the model in order to reduce memory size, processing time, and enhance the accuracy of the model's conclusions. This research provides an implementation of deep learning-based facemask and social distancing detection system.

II.RELATED WORK

Object identification techniques using deep learning techniques have potentially become much more efficient in solving challenging tasks in recent years when compared to shallow models [10]. Developing a real-time system/model capable of recognising whether persons have worn a mask or not in public places is one of the use cases.

Real-time deep learning was used by Shaik and Ahlam [7] to categorize and distinguish emotions, whereas VGG-16 was used to classify seven faces. Under the existing Covid-19 lock-down period for avoiding spreading, this technique thrives. Furthermore, Ejaz et al. [12] employed main component analysis to distinguish people with masked faces from people with unmasked faces.

One use of face recognition was done by Li et al. They created the HGL approach, which used color texture analysis in pictures and line portraits to identify head poses using masks for faces.[11] In an effort to track and enforce compliance with health guidelines, utilizing CNN (Convolution Neural Network) to distinguish whether an individual was wearing a mask or not. The precision in the front was 93.64 percent, while the side-to-side accuracy was 87.17 percent.

Using the condition identification approach, Qin and Li [3] created a face mask recognition design. The four main components of approaching the problem are, p re - processing the image, trimming the facial regions, super-resolution operation, and estimating the end state.

III. METHODOLOGY

The suggested face mask system is implemented utilizing the YOLO v3 algorithm. Deep learning is generally implemented using both artificial intelligence and machine learning techniques. When compared to machine learning, this method, which is inspired by brain neurons, has shown to be more flexible and develops more accurate models. The suggested system comprises two sets of images, one of which is not wearing a mask on the face and the other of which is wearing a face mask. This is done in real time in a specific distance utilizing a web camera to capture and recognise photos. The outcome will be saved in the Internet of Things, which will aid in the process' organization.

The study uses YOLO to detect the wearing and not wearing of face masks in real-time using an alarm system, as well as the observation of social separation among people strolling in the area. A VGG-16 model for detecting face masks that is both precise and fast. The trained model was able to achieve a 97 percent accuracy rate. In YOLOV3, an image is divided into grids first. Around highly scoring objects with the aforementioned specified classifications, the number of anchor boxes (also known as boundary boxes) for each cell will be predicted.

First the image will be processed before further implementation which is called preprocessing. For the necessary detection of boundaries and objects, image segmentation techniques are used. The relationship must be known in order to classify the objects and different categories. To achieve this, the model must be trained with the proper data set. Here we are also using deep learning technology along with OpenCV and computer vision.

The camera should be calibrated in order to match the pixel distance in the image captures. This can be achieved by calibrating it according to required internal and external parameters. Even though it produces accurate results, this one is harder to implement than the other available method which is implemented using triangle similarity calibration but with lesser accuracy.

IV. WORKING

A. YOLOv3 and Dataset

For the effective face mask detection application, we are using YOLOv3 which is the efficient one, and commonly used for the real time object detection systems. As compared to other available techniques this one is faster and accurate. YOLO consists of 53 layers which are trained with ImageNet. For our specific application we are stacking 53 more layers, which combinedly gives 106 layers which we can use for our application of face mask detection. For training the model the required dataset is obtained from the internet. Various data formats such as png, Jpeg are included in the dataset. There are lots of datasets available including Kaggle's dataset for Face Mask Detection by Omkar Gaurav. All the photos were from open-source resources, out of which some resemble real-world scenarios, and others were artificially created to put a mask on the face.

Omkar Gaurav's dataset gathered essential pictures of faces and applied facial landmarks to find the individual's facial characteristics in the image. Major Facial landmarks include the eyes, nose, chin, lips, and eyebrows. This intelligently creates a dataset by forming a mask on a non-masked image. Finally, the dataset was divided into two classes or labels. These were 'with_mask' and 'without_mask', and together, the images were curated, aggregating to around 4000 images.

The quality of the dataset impacts the accuracy of a model.The first step in the data cleanup process is to delete all the incorrect images found in the data set.The photos are downsized to a set size of 96 × 96 pixels, which reduces the strain on the machine while training and ensures the best possible outcomes. Following that, the photos are labeled as having masks or not. To accelerate processing, the photo array is converted to NumPy array.Also used is the MobileNetV2's preprocess input function. The data augmentation technique is then used to expand the size and quality of the data sample. To create multiple copies of the same image, we use the Image Data Generator feature with the required settings for rotation, zoom, and flip horizontally or vertically.

To avoid over - adjust, the amount of training samples has been increased. This improves the generalization and robustness of the formed model. By randomly selecting photos from the dataset, the dataset is then segregated into training and test data in a ratio of 8:2.

In training and test datasets, the stratification variable is used to maintain the same proportion of data as in the original dataset. We used Google Colab to carry out this work. The model was formed on Colab with a GPU, while the preprocessing steps were created on a laptop because they were not computationally intensive.

To form our model, we need to transform the.xml files into.txt and, more specifically, produce the YOLO format after downloading the dataset.Of course, before we begin training, we need to be certain that the conversion has been successful and that we will provide accurate data to our network.

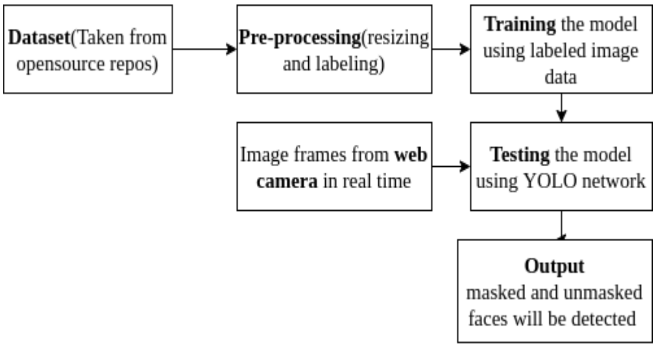

Fig.1 Proposed system block diagram

B. Composition of the Dataset

We have to break our data into two sets, the training and validation sets, to form and validate our model during the training process. The percentages were 90% -10% in each instance. So we made two new files, one for the test folder and the other for the train folder, and placed 85 photographs with their accompanying annotations in the test folder and the remaining 768 into the train_folder directory.

C. Darknet and Required files

Darknet model is used for this specific face detection case. To use this for detection, the additional weights which are present in the YOLOv3 network are randomly initialized prior to the training. They will be getting their proper values during the training phase. We need to create 5 files in order to complete our preparations and start training the model.

1. **sample.names file**: A file with *.names extension* which has all the three classes specific to our problem: *mask, no_mask*, and *not_proper.*
2. **sample.data**: A *.data* extension file which includes relevant information for the problem which will be used for further training.
3. **fmask.cfg**: We need to copy the yolov3.cfg and rename it into a .cfg file for setting our required configuration.
4. **text files for train and test** : Path to all individual files are included in these two .txt files.

Face can be discovered in both static images and real-time video streams using this face detection algorithm. Overall, this model is quick, precise and resource-efficient. Now the model has been trained, it can be used to detect the existence of masks in any image. The given image is initially fed into a face detection model, which detects all of the faces in it. Subsequently, these faces are integrated into an CNN-based face mask detection algorithm. The model would identify the hidden patterns/features of the picture and categorize it as "Mask" or "No Mask."

D. Steps followed

The steps which are followed during the entire process are shown in Fig.1. After getting all the required files and data the **preprocessing** step will happen where we'll be changing the parameters of the image such as dimensions as per our requirement and will convert it into an Numpy array. After that the pixel intensities will be scaled. The obtained image data will be reduced to more manageable groups by the process of **feature extraction**(Fig.2) carried out by the deep learning model we are using. The real time image frames captured from the camera will be processed by the **trained model** to produce the **output.**

E. Social Distancing Monitoring

The following are the stages to develop a social distancing detector.

1. Only persons in an image are detected using object recognition.
2. The **Pairwise distance algorithm** is used to determine the distance between the persons after all of the people in the image have been identified.
3. According to the acquired results, the distance criterion will be violated if any two people are less than N pixels apart, which is the threshold that we set earlier.

Let's imagine there are at least two people in the frame. We'll calculate the **Euclidean distance** between all parts of the centroid. AS the matrix is symmetrical, we have to loop over the upper triangular distance matrix and examine if the distance exceeds the government's and the health care professional's minimal social distance requirements.

V. RESULTS

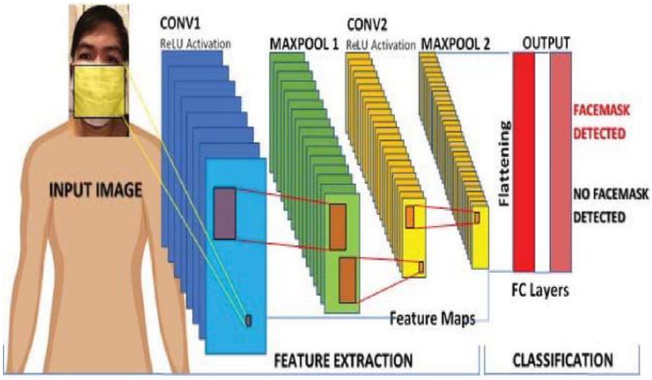

Fig.2 YOLO Model

If a person is not wearing masks, then the boundary box displayed over the face will be red or yellow in color. Otherwise it will be green. This one is illustrated in Fig.3 Similarly, the people who are violating social distancing will be marked in red in the real time image as shown in the Fig.4

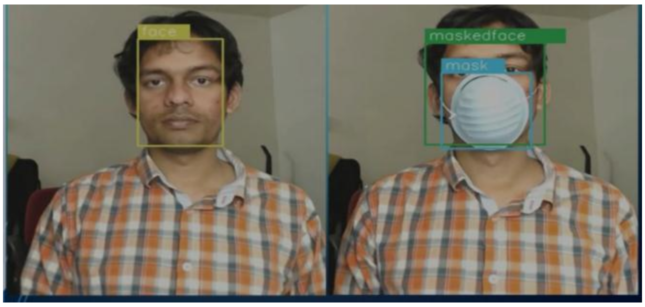

Fig.3 EXPERIMENTAL RESULTS FOR FACE MASK DETECTION(GREEN-MASK DETECTED, YELLOW - NOT WEARING THE MASK)

Fig.4 EXPERIMENTAL RESULTS FOR SOCIAL DISTANCE DETECTION(GREEN-NO VIOLATION,RED- VIOLATION)

VI. CONCLUSION

While the COVID-19 outbreak poses a number of the major hazards to the world, it also serves as a reminder that we must take care to prevent the virus from spreading. By analyzing the photos, YOLO-based techniques proved their ability to recognise and classify objects. The suggested system can detect the presence or absence of a mask, social distancing, and the model delivers accurate and quick results using queries, OpenCV, and CNN. With regard to the speed of detection and the accuracy of the model, the proposed scheme yields admirable results.

REFERENCES:

[1] Viola, P., Way, O.M., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vision 57(2), 137–154 (2004)
[2] Forsyth, D.: Object detection with discriminatively trained part-based models. IEEE Trans.Pattern Anal. Mach. Intell. 32(9), 1627–1645 (2010).

[3] Ning X, Duan P, Li W, Zhang S (2020) Real-time 3d face alignment using an encoder-decoder network with an efficient deconvolution layer. IEEE Signal Process Lett 27:1944–1948. https://doi.org/10.1109/LSP.2020.3032277

[4] Girshick, R.: Fast R-CNN. Comput. Sci. (2015) [5] Ren, S., He, K., Girshick, R., et al.: Faster R-CNN: towards real-time object detection with region proposal networks (2015)

[6] Liu, W., Anguelov, D., Erhan, D., et al.: SSD: Single Shot MultiBox Detector. In: European Conference on Computer Vision, pp. 21–37. Springer, Cham (2016)

[7] Hussain SA, Al Balushi ASA (2020) A real time face emotion classification and recognition using deep learning model. In: Journal of physics: Conference series, vol 1432. IOP Publishing, Bristol , p 012087

[8] Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: 2017 IEEE Conference onComputer Vision and Pattern Recognition (CVPR), pp. 6517–6525 (2017)

[9] Redmon J, Farhadi A.: YOLOv3: an incremental improvement (2018)

[10] Adusumalli, H., Kalyani, D., Sri, R. K., Pratap Teja, M., & Rao, P.B.R.D.P. (2021). Face Mask Detection Using OpenCV. 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV). doi:10.1109/icicv50876.2021.9388375

[11] Li S, Ning X, Yu L, Zhang L, Dong X, Shi Y, He W (2020) Multi-angle head pose classification when wearing the mask for face recognition under the covid-19 coronavirus epidemic. In: 2020 International conference on high

performance big data and intelligent systems. IEEE, pp 1–5. https://doi.org/10.1109/HPBDIS49115.2020.9130585

[12] Ejaz, M.S., Islam, M.R., Sifatullah, M., Sarker, A.: Implementation of principal component analysis on masked and non-masked face recognition. In: 2019 1st International Conference on Advances in Science, Engineering, and Robotics Technology (ICASERT). pp. 1–5 (2019)