A New Technology to detection face by using two Algorithms model

Zeena Mohammed[1] and Dr. Hayder Hassan Safi[2]
[1]Computer Department , Informatics Institute for Postgraduate Studies
[2]Computer Department, College of Basic Education, Mustansiriyah University, Iraq

Abstract:
In the time of Corona and others, many recent studies dealt with revealing the face when wearing a mask, but they lacked the possibility of identifying the face and detecting it with high accuracy for many reasons, including not wearing the mask correctly, the roundness of the face, high lighting, and videos.which negatively affected the accuracy of face identification and identification in many cases.They presented in this study a method that two algorithms, CNN and Yolo V5, in a way that can deal with face detection problems using data bases that are classified into three classes: with mask, without mask, and incorrectly mask. From photo and videos,In the results CNN (Accuracy 0.94 , precision 0.92 , and Recall 0.90), and the results Yolo v5 (Accuracy 0.93 , precision 0.95 , and Recall 0.90).
Keywords: COVID-19, mask in face detection, CNN, Yolo v5, Deep learning , Open CV, Keras,
DOI: 10.24297/j.cims.2023.5.7

1. Introduction

Deep machine learning algorithms are important across the board today, where they have proven their worth in recognizing the faces of people. The initial identification of a coronavirus-infected patient occurred in December of the current year. The COVID-19 pandemic has emerged as a global health crisis [1]. Human beings are ubiquitous. The global situation is currently unstable due to the ongoing pandemic. On a daily basis, a significant proportion of individuals are exposed to pathogenic agents and experience adverse health outcomes as a consequence. As of the time of publication, it was reported that there were roughly 16,207,130 cases of contamination and 648,513 fatalities. This information is cited from reference [2]. The aforementioned metric is exhibiting a gradual increase. According to the World Health Organization (WHO), the most common signs of a coronavirus are fever and a dry cough. Fatigue, diarrhoea, and loss of taste and smell [3] Numerous regulatory bodies have enforced the utilization of facial coverings, particularly in communal settings, while certain nations have implemented COVID-19 limitations such as a nationwide confinement, curfews, travel constraints, the cessation of communal areas, social distancing, and border closures. Extensive contact with people is frequent and inevitable, especially on public transportation and at sports facilities, squares, shopping center's, and workplaces. However, compliance and adherence to the proper wearing of face masks It was difficult for various reasons, including diverse mask types and different degrees of obstacles. There are many things to consider, like balancing the accuracy of different models or errors, deployment needs, and angle. view and publish the detection model on computers with limited processing power, low-resolution images and faces, and a lack of a real data set. The objective of this investigation was to furnish a thorough evaluation of artificial intelligence models that were employed for the purpose of identifying face masks [4]. Traditional biometric systems based on passwords or fingerprints are no longer secure (facial recognition is more secure because it does not require the user to touch any device). Wearing a mask The cause of the following problems is the face: Thieves and crooks wear masks to steal and commit crimes undetected. And when a large part of society needs access control and face authentication, these functions become very challenging. As a result, recognizing face masks is challenging. And due to the spread of the Corona virus, it has received a lot of attention lately [5]. The main problem is recognizing the faces of people who wear the mask in public places, and it depends on the contrast in the size and colour of the faces. Multiple model frames for detecting masked faces and regular faces are proposed, and the available data sets are used to sort them into groups. The process of face mask detection involves the identification of individuals in a group photograph and the determination of whether or not they are wearing a mask. The aim of this study is to concentrate on the identification and acknowledgement of facial masks within a group of individuals, utilizing surveillance footage captured in public areas. In conventional image classification tasks, the images or videos utilized for both training and testing purposes are typically frontal-facing and of comparable dimensions. The recognition of faces in a crowd by a classifier can be affected by variations in the size and resolution of the faces, as noted in reference [6].

2. Related work

As of January 20, 2021, the uncontrollable coronavirus disease 2019 (COVID-19) had spread to 213 countries and territories around the world, as well as two international conveyances, resulting in 96.1 million confirmed cases and 2.06 million deaths worldwide. The absence of resistance and a lack of dynamic medicinal specialists The susceptibility of the people was built up against COVID-19. This was classified as a pandemic by the World Health Organization (WHO) [7]. The only way to deal with reducing the risk of virus corona is to wear the mask. It has become necessary and obligatory to wear a mask, and because a mask makes most of the features of the face disappear, it has become necessary to work on the face detection systems that you want to wear as a mask in terms of security and in terms of the development of information technology, because face detection systems do not need to touch devices but rather just look at them, thus reducing contact and the spread of the epidemic [8]. In this study, we look at past research that shows how important deep machine learning is for finding people who are wearing masks to protect themselves from diseases like the Corona virus.

Deep learning, OpenCV, Tensor Flow, and Keras are used in this paper by S. Shivaprasad. Face detection was aided by the employment of masks as part of a study technique. With the help of this technology, safety is maintained. Face detection was performed using the MobileNetV2 and CNN frameworks. It's a low-cost, low-parameter classifier that can be utilised in embedded devices (Onion). Omega2 and the Raspberry Pi are utilised to accomplish authentic mask identification. The precision of the approach This study's F1-score is 0.92, whereas the one used in this study is 0.96. The data was gathered from a number of different sources. It can be used by a variety of scientists and sources to develop more advanced models, such as face recognition, facial patterns, and facial features for detection. The accuracy result for the mask face dataset (Real-World Masked Face Dataset (RMFD)) is 0.9896 [9].

The deep learning model for face mask identification is studied and developed in this study by J. Ieamsaard, S.N. Charoensook, et al. The YoloV5 uses five distinct numbers of epochs to train the model. When compared to the 86 photos that were examined, the deep learning model for face mask detection with 300 epochs performed the best, with an accuracy of 96.5 percent and the highest precision and recall [4].

To accomplish this objective, S. Singh and colleagues used two state-of-the-art object detection models, namely YOLOv3 and the faster R-CNN. The authors used a collection of photographs of people in two categories—those with and those without face masks—to train both models. This paper suggests a method for drawing bounding boxes (red or green) around people's faces based on whether they are wearing a mask or not and keeping track of the daily ratio of people wearing face masks. In order to compare the performance of both models, we looked at their precision rate and inference time [13].

Fig 1 : Samples from Dataset kaggle including faces without masks ,with Masks, and Mask worn incorrectly [14].

3. Problem statement

As anticipated by experts, the coming 10 years will see the emergence of several new dangerous illnesses. Similar to Coronavirus Disease (19), the best approach for treating these illnesses is to exercise caution, use hand wipes, and wear a facial mask. The user won't get any contagious viruses from donning a face mask, but it will assist in stopping the transmission of disease. Someone can spread viruses into the air whenever they breathe, speak, or sneeze, which might contaminate others.However, the fundamental issue is that people do not recognize the significance of the mask. A deep learning framework has been created to address this issue since it can quickly determine if someone is wearing a mask or not, especially in extremely congested open spaces. Today, using a mask is required; therefore, a camera is installed in each company, so the programmer will identify persons who didn't even use masks and will provide the red indication to perpetrators. And let those who use masks. In the past, a lot of studies have provided complex algorithms for face recognition. In order to learn efficient classifications for recognition and detection, the major study on facial recognition was conducted in 2001. These studies focused on the use of several algorithms, including: CNN and Mobile Net; Mask; RCNN and Yolo v3; transfer learning with Faster-RCNN; and CNN.

There was great suffering among the health and security authorities due to people's negligence in wearing masks. After looking at previous studies, we found that many work on two classes only, but when a face mask is detected in photos or videos, a problem appears, which is the difficulty of detecting the face in different directions due to the rotation of the face, lighting, or other factors. In this research, we provide a method for enforcing the face-mask rule and easily monitoring it in real-time media files. The suggested technology would make it easier and more effective for governments and businesses to regulate face masks.

To address these issues, the hybrid system (CNN, Yolo v5) proposal has been worked on; it works on three classes and is based on face detection in video, not only in pictures, and obtains good results.

4. Methodology

The present investigation employed a research methodology consisting of three distinct stages. The initial phase is comprised of segregated packages that encompass packages such as OpenCV and Keras. The dataset comprises three distinct classes. The study involves the classification of images into three categories: mask (M), without mask (WM), and mask worn incorrectly (IWM). Preprocessing techniques, including data processing and preparation of the data environment, were employed. The second stage encompasses the proposed system, which incorporates detection through the utilization of CNN and Yolo V5. Additionally, the utilization of convolutional neural networks (CNN) and Yolo V5 for data training was implemented. The third stage encompasses performance and comprises the evaluation metric, which denotes the criteria employed in both algorithms.

processing and preparing the data environment

Incorporated packages

Keras For the design and delivery of machine learning (ML) setups with large iteration rates, Keras provides essential insights and building blocks. Layers and models are the two primary data structures in Keras [20]. Keras is used to build each layer in the CNN model. It aids in the construction of the final model and the conversion of the class vector to the binary class matrix in information processing.

OpenCV

OpenCV (Open Source Computer Vision Library), an open-source vision and deep learning software library[19], is used to distinguish and recognize faces, recognize objects, The aforementioned tasks involve the ability to trace progressive modules, monitor eye gestures, track camera actions, eliminate red-eye artifacts from flash photography, retrieve comparable images from a picture database, interpret a landscape, and establish markers to augment it with more realistic details. The proposed approach leverages the OpenCV functionalities for resizing and color correction of image data.

Dataset

In the absence of vaccination, masks represent one of the few preventive measures against COVID-19 and are crucial for safeguarding individuals' health from respiratory infections. By utilizing the data provided on Kaggle [14], Kaggle is a platform that facilitates a network for individuals interested in data science and artificial intelligence. This website is utilized by prominent corporations and organizations to host competitions that offer monetary incentives. Individuals have the ability to disseminate their datasets and peruse previously published data from others while also engaging in competitive events. Furthermore, data scientists have the opportunity to engage in discourse with fellow data analysts within the discussion section and share concise codes that utilize said datasets. Kaggle provides free courses that are accessible to all users, and successful completion of these courses results in the attainment of a complimentary certification. It is possible to construct a model that can differentiate between individuals who are utilizing masks, those who are not adhering to mask-wearing protocols, and those who are wearing masks improperly. Utilizing the Pytorch data within the recommended notebooks The proposed model utilizes a type of Kaggle dataset known as face mask detection. The present assortment comprises encased containers for the 853 photographs stemming from the three courses in PASCAL VOC configuration.

The classifications are as follow:
Individuals wearing a mask (M)
Individuals not wearing a mask (WM)
Individual mask worn incorrectly(MWI)

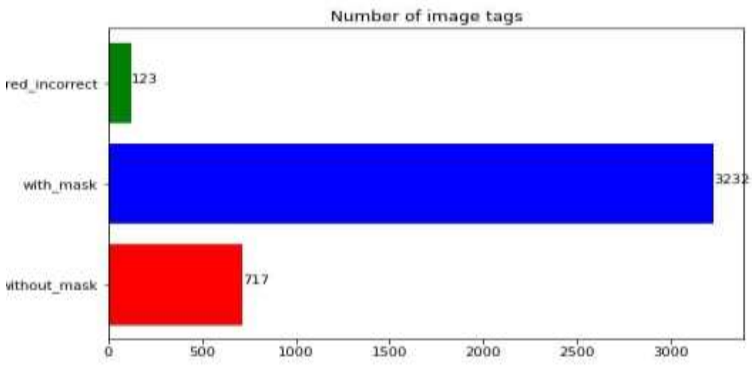


Fig 2: Number of image tags categories

Table 1: Data kaggle [14]

DataSet Name	Main Characteristics	Image Number	Categories ________ M. | WM. |MWI
Face Mask detection	Internet	853	3232| 717 | 123

Data split

Training = 80%

Validation = 20%

Testing = 20%

Table 2: The ratio of the division data of test and train

N.	Training	Testing
With mask	49.15%	47.77%
Without mask	32.23%	33.71%
Incorrect mask	18.62%	18.52%

We can use three color's to cascade the face because the target has three classes.

without mask —> Red

with mask —>Green

mask worn incorrectly —>yellow

Fig3: 3 color's to cascade on the face because the target has three classes

preprocessing

Data Processing

The process of converting data from one format to another, especially one that is more desirable and understandable, is referred to as "data preparation." Any format, including tables, images, movies, graphs, etc., may be used to convey it. These arranged bits of data fit into a composition or model of data, capturing links between diverse objects. The described method works with video and image data using OpenCV.

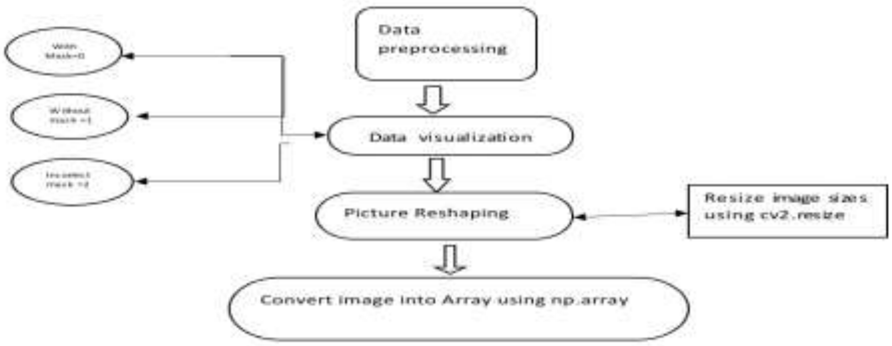


Fig 3:Flowchart preprocessing

Data visualization refers to the transformation of complex data into comprehensible visual representations, which facilitate knowledge sharing and the discovery of novel insights through encoding techniques. It is recommended to conduct a focused investigation on a particular trend that is evident within the dataset. The complete dataset is visually presented in three distinct categories, namely "with mask," "without mask," and "mask worn incorrectly." The statement pertaining to the data path serves to categorize the series of directories comprising the provided data path. The current enumeration of variable categories includes "with mask," "without mask," and "mask worn incorrectly." Subsequently, by utilizing the labels = [i for I in range(len(categories))] approach, it is imperative to partition the aforementioned categories to obtain the count of labels. The assigned values to the labels are [0, 1, 2].

The label dictionary is generated by applying the function dict(zip(categories, labels)) to map each category to its corresponding label. The initial step of this function involves generating an iterator consisting of tuples in the form of a zip object, wherein the elements of each supplied iterator are paired in a corresponding manner. The variable label dictionary for the mapped variable indicates the following: "with mask" is assigned the value of 0, "without mask" is assigned the value of 1, and "mask worn incorrectly" is assigned the value of 2.

Grayscale image generation from an RGB image:The process of producing a grayscale image from an RGB image Contemporary algorithms for image recognition that rely on descriptors often process grayscale images but do not provide an account of the methodology employed for the conversion from color to grayscale. The reason for this is that the utilization of potent descriptors undermines the significance of the color-to-grayscale methodology. The quantity of training data required to achieve optimal performance may increase as a result of the inclusion of extraneous data. The utilization of grayscale is preferred for the extraction of descriptors as opposed to the direct processing of color images, as it streamlines the methodology and diminishes the computational demands. The process of altering the color space is accomplished through the utilization of the cv2.cvtColor (input image, flag) technique. The flag denotes the type of conversion. The conversion to grayscale in this case involves the utilization of the flag cv2.color BGR2gray. Convolutional neural networks with a deep architecture require an input image with a fixed dimension. Therefore, a consistent and uniform size is necessary for every image within the collection. The cv2.resize function was utilized to increase the size of the grayscale image to 128x128

Picture Reshaping: In the process of downsampling an image, a texture tensor is utilized as the input, where each channel comprises a unique and prominent pixel. It is imperative that every image maintain uniform dimensions in relation to the 3D feature tensor. However, it is common for the feature tensors and images to not be entirely congruent. Most convolutional neural networks (CNNs) have a restriction that only permits the use of manipulated images. This gives rise to several challenges pertaining to the acquisition of data and the execution of models. In order to circumvent this limitation, it is possible to make alterations to the input images prior to their enhancement through the neural network. The normalization process is employed to standardize the pixel range of the photographs within the interval of 0 and 1. Subsequently, these entities undergo a transformation process that converts them into matrices with four dimensions.

proposed method

Detection using CNN

After dividing the data, we take the training part to train the model . The process of training a model is the process of feeding the data into a neural network and letting it learn the pattern of the data, which then pulls out the features of the data. A convolutional neural network contains several layers through which data passes to create a model, and these layers Input, convolutional, ReLU, pooling, and fully connected layers make up the bulk of most convolutional neural networks (the fully connected layer is the same as the conventional neural network). These layers can be stacked on top of one another to form a full convolutional neural network. The convolutional layer and the ReLU layer are often referred to as the "convolutional layer" when discussing real-world applications, so the convolutional layer also goes through the activation function following the convolution operation. In particular, many parameters, including the activation function, the weight w, and the deviation b of the neuron, will be used when the convolutional layer and the fully connected layer perform transformation operations on the input. Both the ReLU and pooling layers carry out a constant-valued operation. In order for the convolutional neural network's calculated classification score to correspond with the label for each image in the training set, the network's parameters in the convolutional layer and the fully connected layer will be trained as the gradient decreases.

Among the features of convolutional neural networks are local receptive fields , sparse weights, and shared parameters. When compared to other neural networks convolutional neural networks are better suited for learning image data because of these three ideas.

The following four layers make up a typical CNN:

Input layer

The input layer, which can be a 128 by 128 grayscale image, represents the input to the CNN. and a 2D, 128x128 matrix is supplied to the network as input. This makes it simpler to visualize spatial relationships.

Convolution layers

The very first layer of a convolutional neural network (CNN) is in charge of extracting features from the input images. In this layer, a MxM-sized filter is applied to the input image through a mathematical process called convolution. A multi-channel, two-dimensional image is produced by applying various filters to the input image. In building our model, 10 filters will be applied to the 128 input images, so it will be our output (128*128)*10 and this layer will be repeated three times, the second time applied to the 64 images ,so it will be our output (64*64)*10 and the third time applied 16 images ,so it will be our output(16*16)*10

Pooling layers

The Pooling layer is in charge of making the Convolved Feature smaller in space. Due to the drastic reduction in dimensions, significantly less computing power is needed to process the data.
Different kinds of pools
standard pool
maximum pools.

After a Convolutional Layer, a Pooling Layer is typically used. The primary focus of this layer is to reduce the complexity of the model by decreasing the size of the convolved feature map. This is achieved by isolating each layer and processing its feature map separately. Depending on the specific mechanism employed, various forms of Pooling operations can be performed.

Dropout

By removing some of the neural network's neurons during training, called a dropout layer, overfitting can be prevented (when a model does well on training data but not on new data).

Fully connected layers (flatten ,Dense)

For a CNN, There is always a fully connected layer at the bottom. We connect each of the nodes from the preceding layer to this completely connected layer because it is in charge of classifying the images.

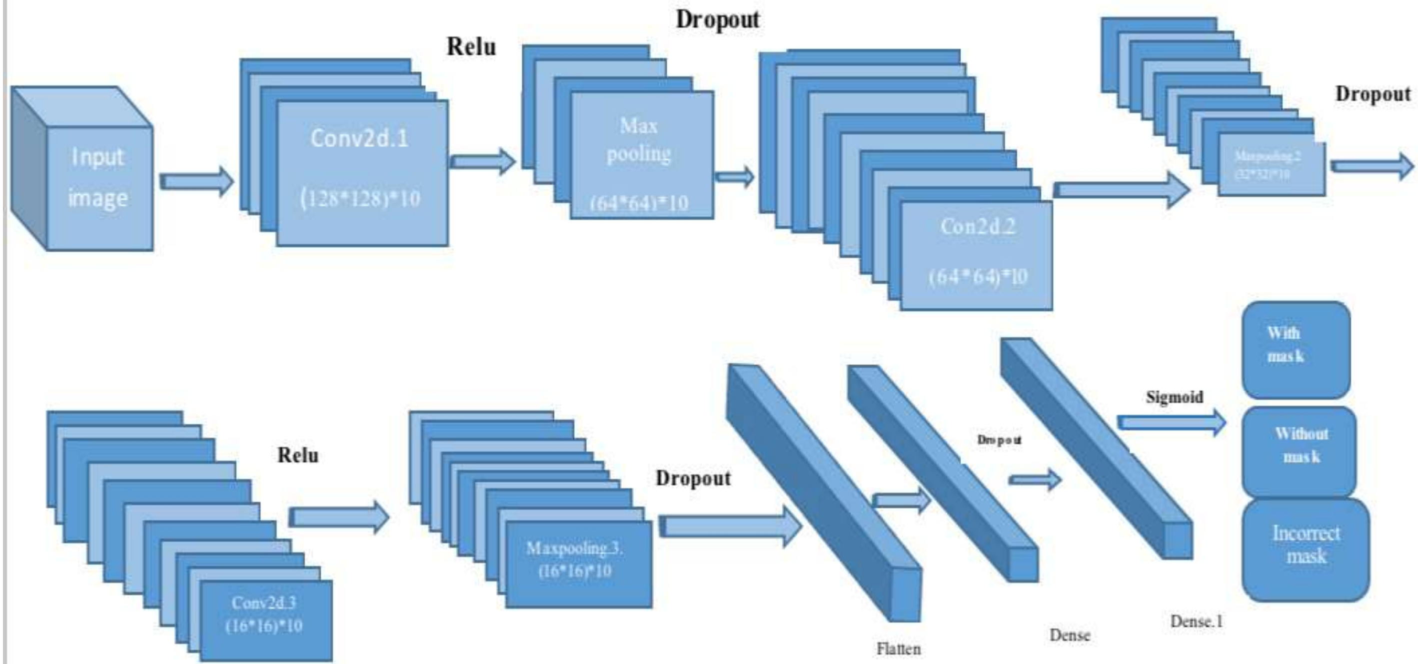


Fig 4: A new architecture of the proposed CNN for model

As shown Figure 4 , a typical architecture of a CNN, which consists of multiple convolution layers followed by a pooling layer. Feature extraction occurs at increasingly abstract levels with each successive convolution and pooling layer. For instance, the first layer's filters could identify both horizontal and vertical edges, as well as diagonal ones. The subsequent layer's filters were shape-detectors, while the final layer's were collection-detectors. The learning algorithm is given an initial set of filter values chosen at random. Because of their ability to perform both classification and feature extraction automatically, CNNs are extremely effective. CNN is unique among classification methods because it can also extract features, unlike methods like Support Vector Machines.

Training CNN

The CNN starts off with completely arbitrary weights. CNN is trained by presenting a large dataset of images(Kaggle)with associated class labels to a neural network (with mask, without mask, Incorrectly worn mask .). The CNN network runs each image through a random value generator and then make comparisons with the class label of the input image.In order to adjust the weight values, a method called back-propagation is used. The tuning process is simplified and improved by the use of back-propagation, allowing for more precise adjustments. Each iteration through the image dataset training process is referred to as a "epoch ".throughout its training process, the CNN undergoes a number of iterations, or epochs, during which its weights are fine-tuned by the necessary incremental changes.

The neural network improves its ability to identify and predict the category of training images with each successive epoch step. Improvements to the CNN result in increasingly minor tweaks to the weights. We use a test dataset to check the CNN's accuracy after training it. The test dataset consists of annotated pictures that weren't used during training. Feeding images into a convolutional neural network (CNN) and comparing the resulting classification with the true class label of the test image. The test dataset is used to assess the CNN's predictive abilities. It is said that a CNN is "overfitting" if its accuracy is high on the training data but low on the test data. This occurs because there is a smaller dataset (training).

Figure 5 ,show split data kaggle into training and testing

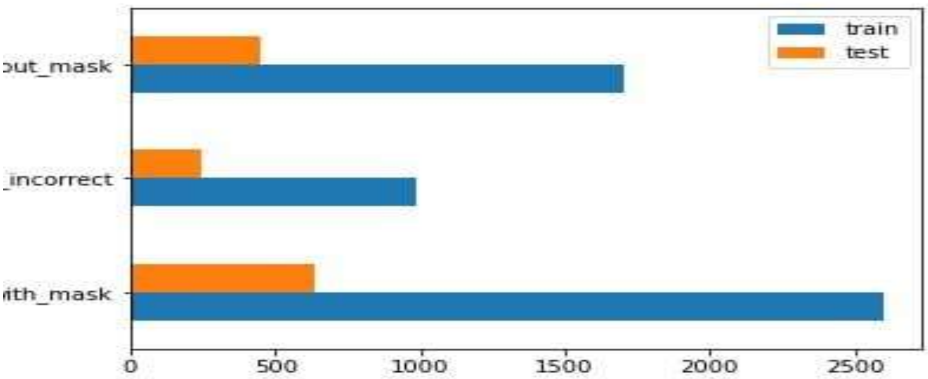


Fig 5: train and test data of CNN

Shown Table 3, Numbers of split data of three classes to Train and Test

Table 3: Train and Test data to three classes of CNN

N.	Train	Test
With Mask	2600	632
Without Mask	1705	446

Mask Incorrect	985	245

Detection using Yolo v5

YOLOv5 is the deep learning framework used to identify and categories GUI elements. Object detection networks have a standard four-part architecture that YOLOv5 mimics: input, backbone, neck, and head. Focus, bottleneck, bottleneck CSP, and the SPP layer are all built on top of CSPDarknet53 in the first iteration of YOLOv5. The bottlenecks that narrow or widen the available channels make up the bulk of the backbone module.

backbone : is a collection of convolutional neural networks (CNNs) that combine features from images with varying levels of granularity.

Neck: It takes image features and mashes them up before forwarding them through a series of layers for prediction.

- **Head**: It uses neck features for prediction of **both bounding boxes and class labels.**

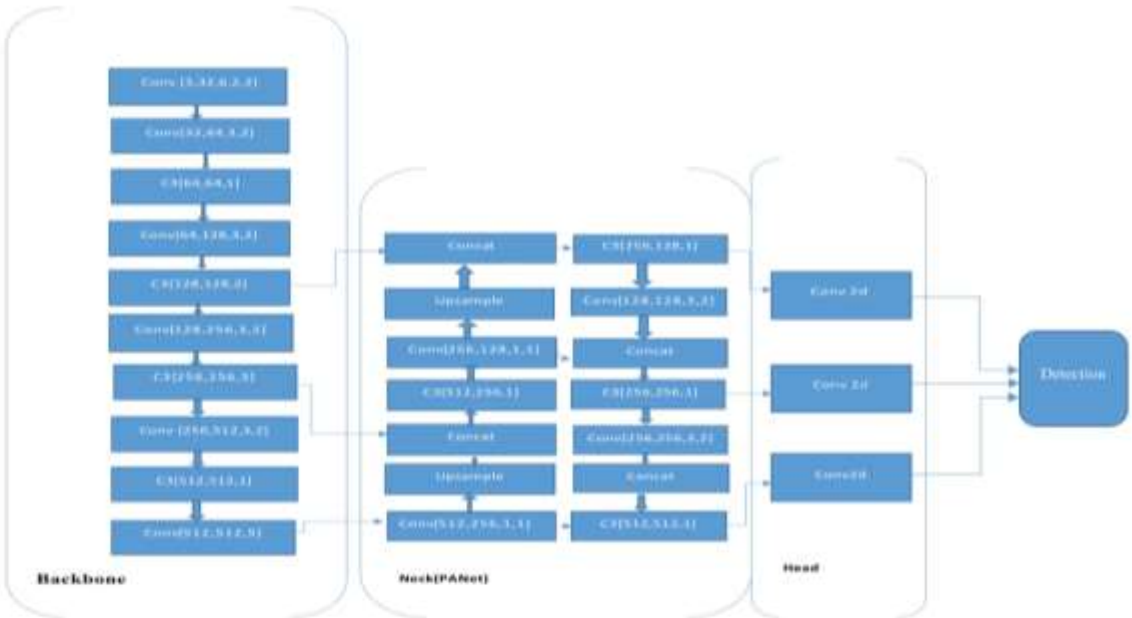


Fig 6 : A new architecture of the proposed Yolo v5 for model

Training yolov5

We used a dataset available at Kaggle to both train our models and guide our experiments. Since there are approximately 853 images and 3 classes (with mask , withoutMask , Mask worn incorrectly), its primary function is to aid in face mask detection (see Figures 7(a))

We randomly divided data into training, validation, and testing sets with 80%,20%, and 20% ratios respectively. Figure 7 depicts the visualization results of the dataset analysis and (a) represents the distribution of object categories in the dataset, (b) represents the distribution of object centroid locations, with the horizontal and vertical coordinates indicating the location of the centroid, and (c) represents the **distribution of object sizes.**

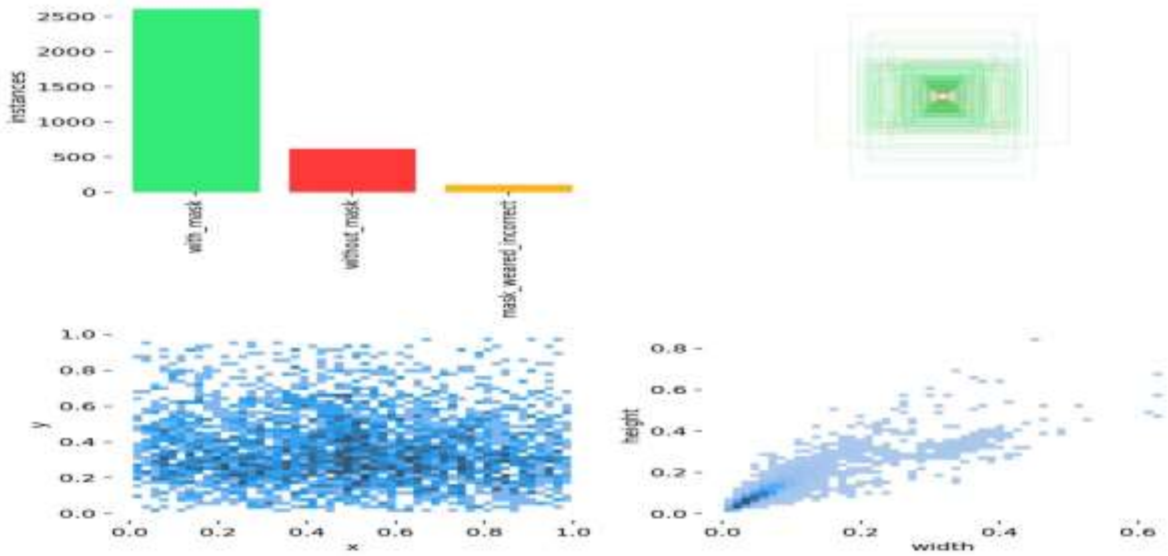


Fig 7: The visualization results and analysis of the dataset

Distributed categories in the data set (M),(WM),(MWI)

the distributed centroid location

the distributed of object size

performance

Evaluation Metrics

The quality of the suggested methodology was assessed using the following four metrics: precision, recall, mAP(mean), andaccuracyThe values of true positives (TPs), true negativ es (TNs), false positives (FPs), and false negatives (FNs) had to be defined in order to be calculate d (FNs).

Accuracy (Acc.) represents the percentage of tests that correctly reported the results (whether positive or negative). Equation (1) provides the formula needed to calculate the accuracy.

Accuracy (Acc.)=TP+TN/TP+FP+TN+FN. (1)

In the precision figure 8, precision is the ratio of positive findings to predictions made by the classifier. Because the precision values are rising across all classes , the detection model is operating effectively. The average level of confidence is 0.955 Thus, precision (Prec.) measures the test' s capability to detect person is wearing a mask

Equation (2) presents the formula for precision.

Precision (Prec.). = TP/TP+FP. (2)

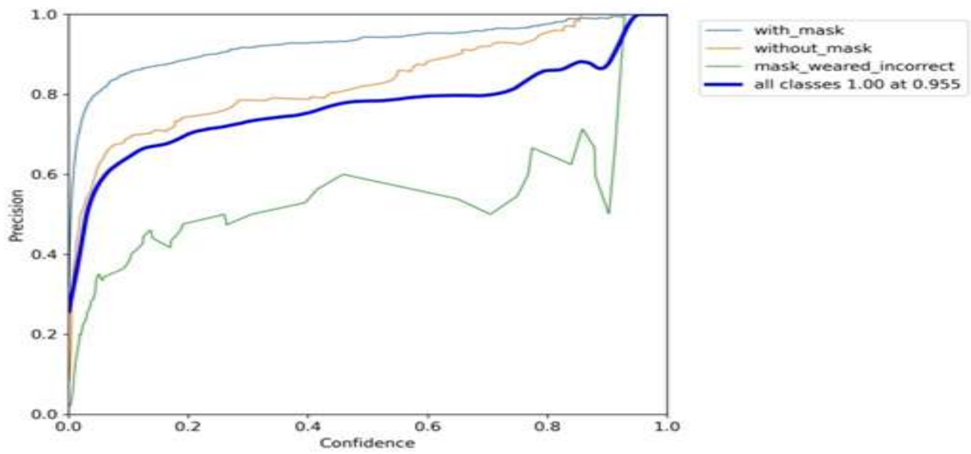


Fig 8: Precision graph of the yolov5 model

Table 4: Precision of Yolo V5 and CNN

N.	Yolo	CNN
With mask	0.96	0.99
Without mask	0.94	0.88
Incorrect mask	0.55	0.88

Figure 9 ,In recall shows the average of the all classes is 0.90

The recall (Rec.), which was calculated using Equation(3)

Recall (Rec.)=TP/TP+FN. (3)

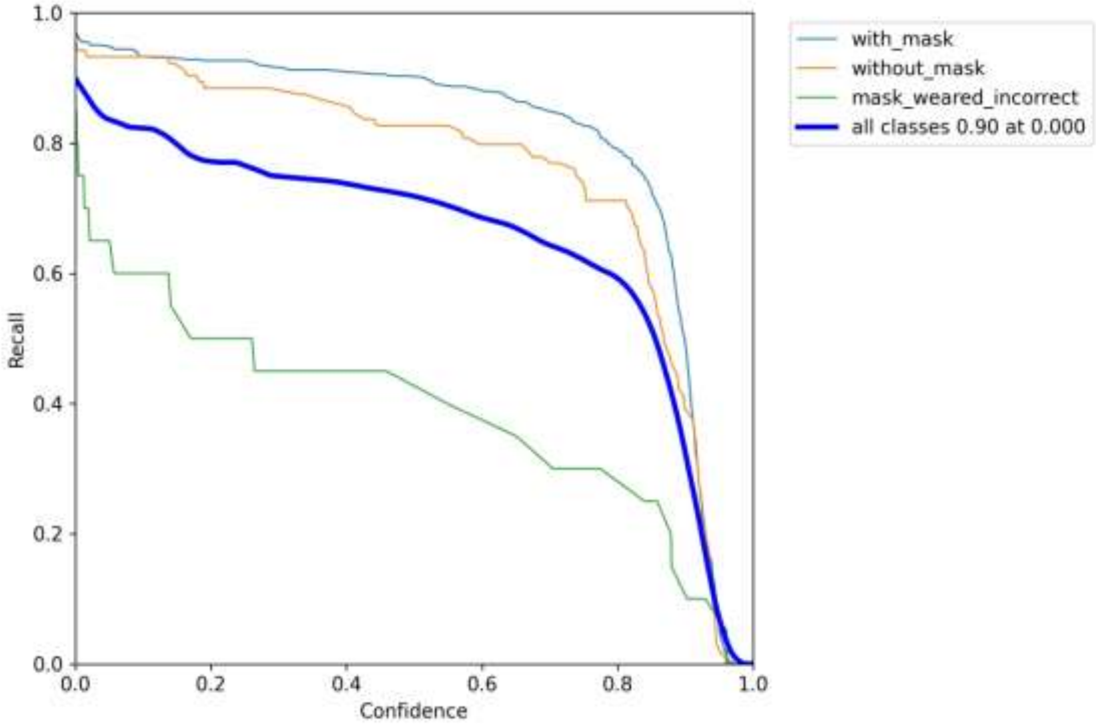


Fig 9 :Recall graph of the yolov5 model

Table 5:Recall of Yolo and CNN

N.	Yolo	CNN
With mask	0.97	0.99
Without mask	0.95	0.95
Incorrect mask	0.77	0.75

Precision and recall (as shown in figure 10), this is a combined graph of precision and recall where with mask value is 0.952, without _mask value is 0.901, mask_weared_ incorrect value is 0.435 , the average values of all classes are 0.763 and mAP value is 0.5 which is a better result.

The Mean Arterial Pressure which was calculated using Equation(4).

mAP=SP+2DP/3. (4)

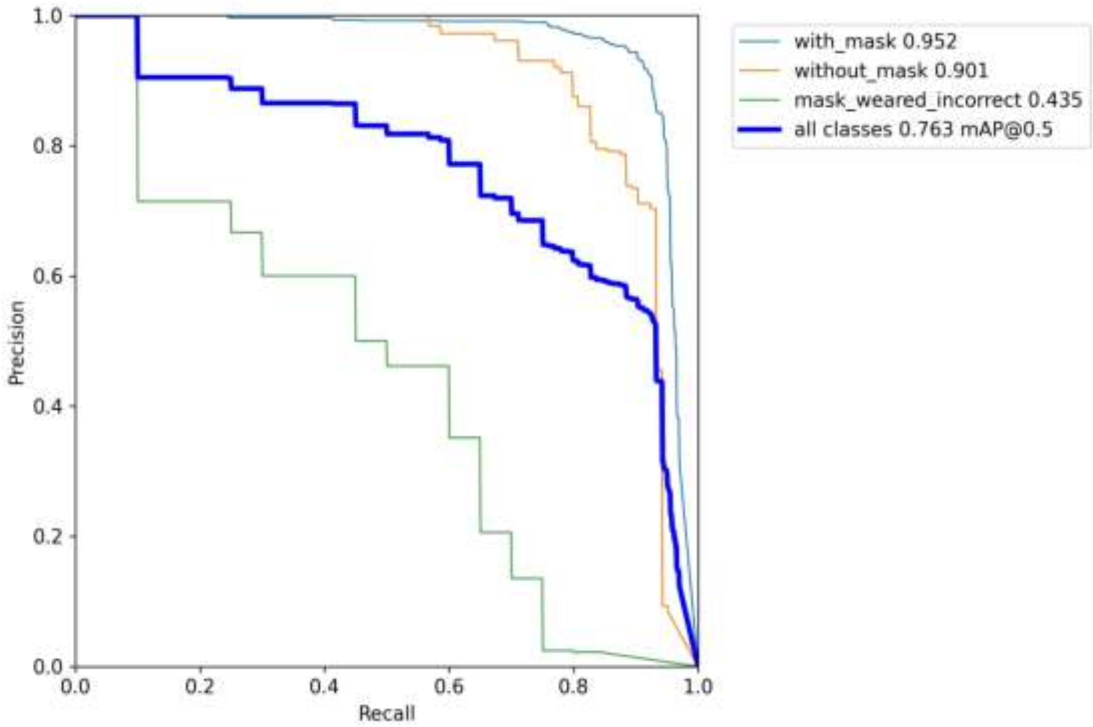


Fig 10: mAP(Mean Average Precision) graph for the yolov5 model

4.3.2 preform proposed method

In this part, we contrast the proposed model's performance on the Kaggle dataset with that of three other cutting-edge face mask detection techniques. Five one-stage models—YOLOv3 [11], Yolo v5 [34]—and two two-stage models—Faster R-CNN [11] and RCNN [15]—make up the three models. Table 6 displays the comparing findings.

Table 6: Comparison of various state of the art methods on the basis of accuracy and precision (pre)and Recall (Rec)

Source	Methodology	Result (%)
[9]	CNN +Mobile Net	ACC= 0.96
[11]	RCNN + yolo v3	Pre=0.90
[15]	Transfer learning With F-RCNN	Pre=0.81 Recall=0.84
Proposed Methods	CNN and yolo v5	Acc CNN =0.94 , Pre0.92 And Recall=0.90 Acc Yolo v5 = 0.93 , Pre Yolo

		v5 =0.95 And Recall =0.90

Upon comparing our model with the initial CNN+MobileNet model, it is observed that the latter yields high accuracy. However, it is limited to only two classes, namely, with mask and without mask. In contrast, our model is capable of working with three classes, namely, with a mask, without a mask, and a mask worn incorrectly.

Upon examination of the second model, it is observed that the precision rate is 0.90. Our models have achieved a high rate of precision when compared to the second model, which utilizes the RCNN and Yolo v3 algorithms, with a precision rate of 0.92 using the CNN algorithm and a precision rate of 0.95 using the Yolo v5 algorithm. Upon examination of the third model, it is observed that the precision rate is 0.81 and the recall rate is 0.84. which utilizes transfer learning with F-RCNN, and our model achieves a precision rate of 0.92 and a recall rate of 0.90 with the CNN algorithm and a precision rate of 0.95 and a recall rate of 0.90 with the Yolo v5 algorithm.

5. Experimental Results and Discussion

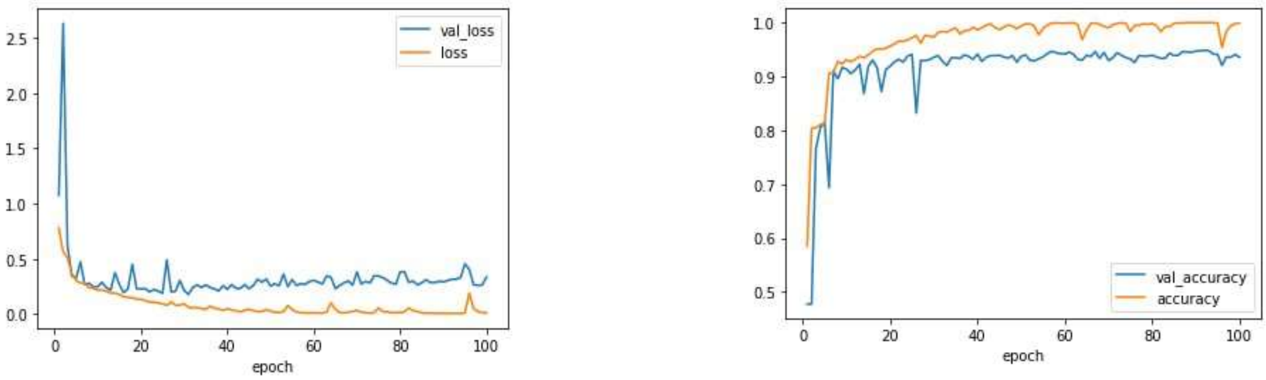
Result and analysis

It is clear from the prior discussion that the efficiency of the face detection model affects the efficiency of the facial mask classifier. The effectiveness of the facial mask classifier is impacted if the face detection model is unable to identify a face or wrongly classifies an item as a face. The following significant findings were found about the efficiency of the suggested strategy:

Testing CNN

The mask detection feature was binary classified by CNN. Image data with a size of 128*128 is used as the training data for 80% and the 20% for validation of the data , the training data divided into 3 classes (with mask: 49.15 %, without mask: 32.23 %, mask weared incorrect: 18.62 %) .

Shown in Figure11, For training loss, validation loss, training accuracy, and validation accuracy, a plot of 100 epochs against loss or accuracy is shown. The plot clearly demonstrates that when the number of epochs increases while the training and validation accuracy decreases, the training and validation accuracy increases. Furthermore, validation accuracy is greater than training accuracy, demonstrating that overfitting has not affected the model. The plot of the number of epochs against the loss or accuracy



a)Loss Curve CNN. b)Accuracy Curve CNN

Fig 11: (a) training and validation loss with CNN.(b) training and validation accuracy with CNN

Fig 12: Deducted Results – without_mask & with_mask & incorrect mask with CNN

Testing Using Yolo v5

The unified architecture of YOLO is a very quick architecture that transforms object recognition from picture pixels to bounding box coordinates and class probabilities into a single regression issue. With this architecture, you can predict what objects are presented where they are by only looking at an image once (YOLO) [15]. Using the YOLOv5m implementation and YOLO format annotations, we trained the data across 100 epochs. Several metrics, including precision, recall, and mAP (mean average precision), where IOU are between 0.5 (or 50%) and 0.95 (95%), were used to evaluate the model's performance after training.

The model's performance on training and validation data is shown in Fig13. In the training and validation data (top row and bottom row, respectively), the first three columns show the bounding-box loss, object loss, and classification loss. As the number of training epochs increases, we can see that the precision and recall are likewise increased in the validation data. The object's location is also provided by YOLO in the form of a bounding box, which may be verified using IoU.

The graphs of the metrics curves as training advances are shown in Figure 6. Following After evaluation, the YOLO model had a validation precision score of 0.95, a recall score of 0.90, and mAP scores of 0.76 and 0.45 for @0.5IOU and @0.95IOU, respectively.model had a validation precision score of 0.8057, a recall score of 0.95, and mAP scores of 0.95 and 0.64 for @0.5IOU and @0.95IOU, respectively. This result confirms the effectiveness of our approach in correctly predicting signs performed in diverse environments.

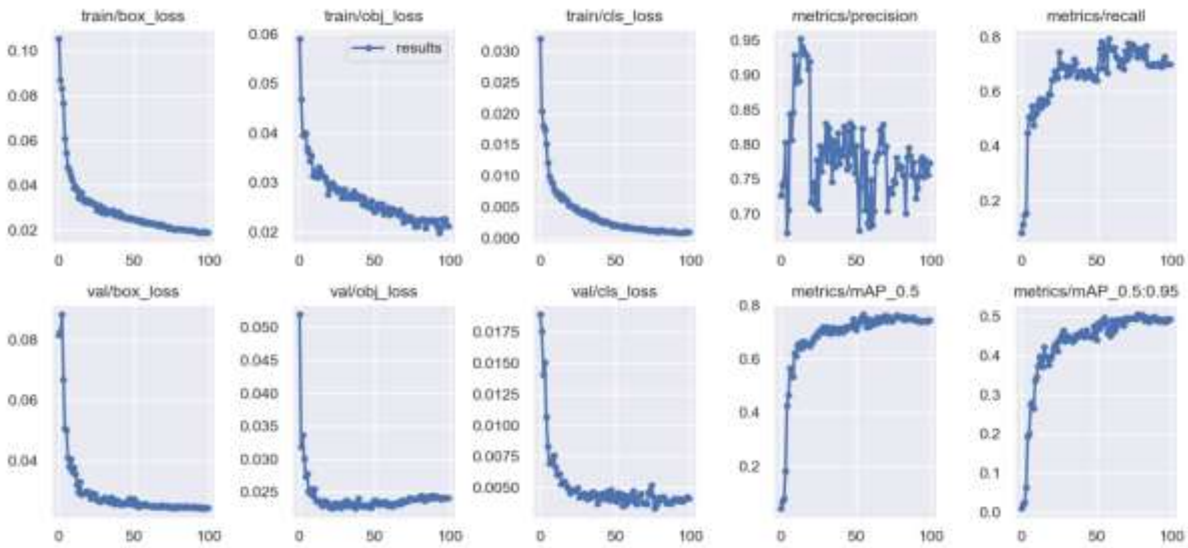


Fig 13 :Graph of Precision, Recall, and mAP as YOLOv5 training progresses.

Table 7: performance and results both for yolo v5 and CNN

Model	Accuracy	Precision	Recall
CNN	0.94	0.92	0.90

Yolo v5	0.93	0.95	0.90

6. Conclusions and future work

We conclude that the challenge of wearing a mask on your face is one of the most difficult, as about half of the face is missing, leaving only the eyes and forehead. From a security point of view, it is difficult to distinguish a face, and most detection methods rely on contact, such as a fingerprint or a vein, which is dangerous to touch from an epidemiological point of view. This problem was concluded based on research conducted between 2015 and 2022. Our research presents previous studies in this field in order to be able to access and display the strengths and weaknesses of all models and to enable us to improve work on these models using deep learning algorithms that have proven their merit and effectiveness in the detection of people wearing the mask during the past years.

As a result, the scientists have suggested an integrated method for identifying face masks on people. Concerns about face mask recognition are crucial to stopping the spread of COVID-19. The suggested study was completed in two stages. The first stage involved classifying face masks using CNN, and the second stage involved using the YOLOV-5 algorithm. As a first stage, using a data set Kaggle divided into three classes—with mask, without mask, and mask incorrectly—a single frame from an openCV movie is retrieved. In order to determine if humans were present and to generate bounding boxes around each one, it was then processed using YOLOv5. The boxes were then sent to be face-scanned after that. The faces were then sent to CNN and Yolov5 for facial mask recognition. The suggested CNN excelled in our comparative investigation for face mask identification, and it served as the basis for our Face Mask Detection method. The YOLOv5 has a mAP (mean average precision) of 0.763 and a precision and recall of 0.95 and 0.90, respectively.

This technology may be further enhanced by utilising several cutting-edge face recognition algorithms. The face recognition task may be enhanced by using additional transfer learning algorithms and methods, such as hyper parameter tuning. A less reliable model for the testing pictures may be obtained by using a more varied dataset. Better results may be obtained by combining several state-of-the-art algorithms with various data augmentation approaches. The results of the research may simply be integrated with CCTVs in public areas, stopping the spread and saving lives. By giving the mask classifier real-time CCTV facial data to train on, this method may be made more effective. It may also be used in conjunction with hardware to alert authorities as soon as congestion worsens so they can take the appropriate action. Since the world has already experienced a pandemic, this strategy may be utilised to combat any future epidemics since these two actions can significantly slow the spread of the disease. In order to make our research more useful, we'll concentrate on how well tiny faces can be detected in crowded public settings.

References

1. H. Lau et al., "Internationally lost COVID-19 cases," J. Microbiol. Immunol. Infect., vol. 53, no. 3, pp. 454–458, 2020, doi: 10.1016/j.jmii.2020.03.013.
2. "COVID Live - Coronavirus Statistics - Worldometer." https://www.worldometers.info/coronavirus/ (accessed Jun. 04, 2022).
3. "Coronavirus disease (COVID-19)." https://www.who.int/emergencies/diseases/novel-coronavirus-2019?gclid=Cj0KCQjwheyUBhD-ARIsAHJNM-OBSN3eJiC8nxnHHf6Ym7Qt1tBcn8-mroRXwBLGfwhZj0BEvPnmMqQaArtJEALw_wcB (accessed Jun. 04, 2022)
4. J. Ieamsaard, S. N. Charoensook, and S. Yammen, "Deep Learning-based Face Mask Detection Using YoloV5," Proceeding 2021 9th Int. Electr. Eng. Congr. iEECON 2021, pp. 428–431, 2021, doi: 10.1109/iEECON51072.2021.9440346.
5. F. M. J. Mehedi Shamrat, S. Chakraborty, M. M. Billah, M. Al Jubair, M. S. Islam, and R. Ranjan, "Face Mask Detection using Convolutional Neural Network (CNN) to reduce the spread of Covid-19," Proc. 5th Int. Conf. Trends Electron. Informatics, ICOEI 2021, pp. 1231–1237, 2021, doi: 10.1109/ICOEI51242.2021.9452836.
6. S. Liu and S. S. Agaian, "COVID-19 face mask detection in a crowd using multi-model based on YOLOv3 and hand-crafted features," no. April 2021, p. 20, 2021, doi: 10.1117/12.2586984.
7. "The World Health Organization." https://www.gavi.org/operating-model/gavis-partnership-model/who?gclid=CjwKCAiA9aKQBhBREiwAyGP5labQ3hE9jhAL9lkh66XMFeETSrj9_ofmoGfj6XS6oJIjUcOrIxMdehoCd2oQAvD_BwE (accessed Feb. 13, 2022).
8. S. Chaudhry and R. Chandra, "Face detection and recognition in an unconstrained environment for mobile visual assistive system," Appl. Soft Comput. J., vol. 53, pp. 168–180, 2017, doi: 10.1016/j.asoc.2016.12.035.

9. S. Shivaprasad, M. D. Sai, U. Vignasahithi, G. Keerthi, S. Rishi, and P. Jayanth, "Real Time Cnn Based Detection of Face Mask Using Mobilenetv2 to Prevent Covid-19," vol. 25, no. 6, pp. 12958–12969, 2021.
10. A. Chavda, J. Dsouza, S. Badgujar, and A. Damani, "Multi-Stage CNN Architecture for Face Mask Detection," 2021 6th Int. Conf. Converg. Technol. I2CT 2021, 2021, doi: 10.1109/I2CT51068.2021.9418207.
11. J. R. Sella Veluswami, S. Prakash, S. C. B, and N. Parekh, "Face Mask Detection Using SSDNET and Lightweight Custom CNN," SSRN Electron. J., no. January, 2021, doi: 10.2139/ssrn.3882472.
12. G. Yang et al., "Garbage Classification System with YOLOV5 Based on Image Recognition," 2021 6th Int. Conf. Signal Image Process. ICSIP 2021, vol. 1, no. January 2020, pp. 11–18, 2021, doi: 10.1109/ICSIP52628.2021.9688725.
13. S. Singh, U. Ahuja, M. Kumar, K. Kumar, and M. Sachdeva, "Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment," Multimed. Tools Appl., vol. 80, no. 13, pp. 19753–19768, 2021, doi: 10.1007/s11042-021-10711-8.
14. https://www.kaggle.com/datasets/andrewmvd/face-mask-detection
15. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-timeobject detection. In Proceedings of the IEEE conference on computer vision and patternrecognition, pages 779–788, 2016
16. Y. Fang, Y. Nie and M. Penny, "Transmission dynamics of the COVID-19 outbreak and effectiveness of government interventions: A data-driven analysis", J. Med. Virol., vol. 92, no.6, pp. 645-659, Jun. 2020
17. J. D. Arias-Londono, J. A. Gomez-Garcia, L. Moro-Velazquez and J. I. Godino-Llorente, "Artificial intelligence applied to chest X-ray images for the automatic detection of COVID-19. A thoughtful evaluation approach", IEEE Access, vol. 8, pp. 226811-226827, 2020.
18. YOLOv5 New Version Improvements and Evaluation, Roboflow Blog (2021). https://blog.roboflow.com/yolov5-improvements-and-evaluation/. Accessed 05 Apr 2021
19. "OpenCV", Opencv.org, 2020. [Online]. Available: https://opencv.org./2020.
20. K. Team, "Keras documentation: About Keras", Keras.io, 2020. [Online.[Available: https://keras.io/about. 2020