Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
DTLMV2—A real-time deep transfer learning mask classifier for overcrowded spaces

Meenu Gupta\textsuperscript{a,b}, Gopal Chaudhary\textsuperscript{c,}\textsuperscript{∗}, Dhruvi Bansal\textsuperscript{c}, Shashwat Pandey\textsuperscript{c}

\textsuperscript{a} Department of Computer Science and Engineering, Chandigarh University, Punjab, India
\textsuperscript{b} University Centre for Research & Development, Chandigarh University, Punjab, India
\textsuperscript{c} Bharati Vidyapeeth’s College of Engineering, Delhi, India

Abstract

Through the commencement of the COVID-19 pandemic, the whole globe is in disarray and debating on unique approaches to stop this viral transmission. Masks are being worn by people all around the world as one of the preventative measures to avoid contracting this sickness. Although some people are following and adopting this precaution, others are not, despite official recommendations from the administration and public health organisations has been announced. In this paper DTLMV2 (Deep Transfer Learning MobileNetV2 for the objective of classification) is proposed - A face mask identification model that can reliably determine whether an individual is wearing a mask or not is suggested and implemented in this work. The model architecture employs the peruse of MobileNetV2, a lightweight Convolutional Neural Network (CNN) that requires less computing power and can be readily integrated into computer vision and mobile systems. The computer vision with MobileNet is required to formulate a low-cost mask detection system for a group of people in open spaces that can assist in determining whether a person is wearing a mask or not, as well as function as a surveillance system since it is effective on both real-time pictures and videos. The face recognition model obtained 97.01% accuracy on validation data, 98% accuracy on training data and 97.45% accuracy on testing data.

© 2022 Elsevier B.V. All rights reserved.

1. Introduction

The COVID-19 influenza is a severe abrupt cardiopulmonary disorder coronavirus–caused pandemic that is still going on. The coronavirus outbreak has resulted in unprecedented levels of collaboration with regards to international research. Artificial intelligence based on ML and DL techniques can help combat against COVID-19 in a variety of ways. Scientists and paramedics may use ML tools to analyse reams of statistics to anticipate the transmission of COVID-19, operate as a catastrophic alert system, and identify those who are susceptible. Endowment in formulating AI, machine learning, IoT, and big data is essential to combat and anticipate new ailments. The AI’s abilities are being put to use in the fight against the COVID-19 outbreak, namely the identification of COVID-19 in diagnostic chest X-rays, to explore morbidity and mortality rates [1] and to trace and promptly discover illnesses.

In dealing with the emergence and dissemination of COVID-19, governments are facing several obstacles and hazards. As the virus is spread mostly by minute droplets created by sneezing, coughing, and chatting amongst persons in close proximity. Rather than travelling long distances via the air, droplets usually land on the ground or other objects. Various countries have laws requiring that people wear face masks publicly. These regulations and laws were enacted in response to the rapid rise in incidents and deaths in a range of areas. So far, 7.03 million COVID-19 cases [2] have been reported in over 188 countries and territories, with over 403,000 fatalities. On May 21, there were 100,000 new infections globally [3], the most since the epidemic began, with a total of 5 million people being infected. With a total of 100,000 reported instances on 19 May and 200,000 on 3 June 2020, India presently has the largest proportion of verified cases in Asia. COVID-19 spreads at an exceptionally high reproductive rate, making it difficult to manage. Because no effective 100% vaccination has been completed to fight prevent COVID-19, it is critical to take prophylactic precautions. Wearing facial masks is indeed one of the instructions received from several public health bodies, including the World Health Organization (WHO) [4] and reputed governments, in this current epidemic. The WHO changed its guideline on wearing face masks in crowded places in June 2020, stating that the body needs to be covered specifically the face of a person in public areas to reduce the transmission of this pandemic of COVID-19. In more than 75 nations, wearing a mask is required, and these countries account

https://doi.org/10.1016/j.asoc.2022.109313
1568-4946/© 2022 Elsevier B.V. All rights reserved.
for 88 per cent of the world’s population. Scientists have also discovered that wearing of masks works to minimise viral infection and dissemination. The approach to measuring masses of people, on the contrary, is becoming more complicated. Throughout the surveillance technique, individuals who are not wearing a facial mask are detected. Face mask detection technique looks to be effectively handled because of significant improvements in the field of ML techniques. This technique is increasingly substantial today since it is utilised to recognise faces in real-time evaluation and surveillance as well as in snapshots and videos. Very significant success in picture identification and object detection may be attained owing to enhancement in the CNN model and deep training. Face with the rapid spread of the COVID-19 outbreak, several face-recognition techniques are presently being deployed on people wearing masks.

Masked facial recognition has an accuracy of around 85%, corresponding to Hanvon Technology. Minivision Technology was able to reach a level of accuracy of over 90%. The authors [5] employed the YOLOv3 method to identify facial masks. This approach has a 93.9 percent accuracy rate. The face-eye-based multi-granularity model has a recognition accuracy of 95 percent. The accuracies were reached on a synthetic dataset, which is not the situation in this presented study, which includes both actual and counterfeit photographs. Wearing a face mask correctly has proven to be a good approach to reduce COVID-19 exposure. The perfectly alright wearing condition of the face mask is observed in this work: a face without a mask, a face with the incorrect mask, and a face with the proper mask. The paucity of realistic datasets (containing individuals from all over the world) and the low intra-class distance and massive inter-class distance are the two most difficult aspects of this undertaking. As a solution to the former, a new practical dataset has been built covering numerous classes of mixed-race and polyethnic people, which contains in total 7514 different faces. Many photos have been scraped from public news outlets reporting 2020’s Black Lives Matter movement to bring ethnicity and make this dataset neutral. For the second significant stage, an innovative detection framework has been described to observe the proper detection of wearing face masks, named DTLMV2. This paper consists of 5 sections: Sections 1 and 2 introduce the nature and significance of the study, Section 3 explains the proposed architecture of proposed algorithm, with subsequent Sections 3.1–3.3 explaining the different pre-processing techniques to introduce non-linearity in the data. Sections 3.4–3.5 explains the different layers involved in proposed deep transfer learning model. Section 4 shows the outcomes of the DTLMV2 as performed on the test dataset. The Section 5 discusses the conclusion and future scope of the work.

2. Related work

Face design and identity identification when wearing face masks are the subjects of the majority of the publications. The purpose of the whole research is to find out who have not accessed facial masks in order to prevent COVID-19 from spreading. Scientists and researchers have demonstrated that using face masks reduces of COVID-19 dissemination. Viola-Jones and M. Jones [6] presented the boosted-based cascade architecture with basic yet fast Haar features as one of the earliest and most well-known face detectors. Furthermore, Principal Component Analysis (PCA) was used to create a face mask detector model. In [7], the authors employed PCA on masked and unmasked facial recognition to identify the individual, however, identification accuracy reduces [5] to less than 70 percent when the recognised face is properly masked. In addition to the standard face detectors mentioned above, YOLOv3 and CNN-based models have made significant development in recent years. The YOLO method was employed by the authors in [9] to recognise faces in real-time. Darknet-53 [5] serves as the foundation for YOLOv3. The proposed approach has an accuracy of 93.9 percent. It was given training on approximately 600,000 photographs from the Celebi and WIDER FACE datasets. In the evaluation, the FDDB dataset has been employed. Face detector models in CNN-based categorisation train directly from the operator’s input and then apply numerous deep learning algorithms to it [10]. Further, in 2007, the Cascaded CNN model was introduced [11]. In [12], the author has provided a hybrid model for face mask detection utilising deep and conventional machine learning employing three datasets (Labeled Faces in the Wild (LFW), Real-World Masked Face Dataset (RMFD), and Simulated Masked Face Dataset (SMFD)) in this work. The author presented the Context-Attention R-CNN detection framework in [13], which expands the 2-classes face mask detection issue into a triple-classes detection problem and detects finer-grained situations of wearing face masks. The author [14] suggests a two-stage approach for detecting mask wearers in this paper. It investigates the Faster RCNN methodology using InceptionV2 as a pre-detection stage and BLS as a verification step. Its effectiveness is demonstrated by the combination of two stages. In [15], the face indicator prototype with MobileNetV2 using transmission learning has been trained and features of the face have been extracted from real-time photos and videos using a Haar-feature which uses cascade classifier. The face mask indicator model is then used on the extracted human faces to ascertain whether the individual is wearing a mask. In [16], the authors employed OpenCV DNNs for face mask recognition, which enables present recognition with minimal resource utilisation. It can also recognise faces in various orientations and detect obstructed faces with high accuracy. Several prior models are outperformed by the suggested SSDMNV2 model. In [17], the authors created a novel approach to identifying facemask-wearing conditions. In this way, they able to categorise 3 types of facemasks wearing circumstances. The scenarios are accurate face mask-wearing, wrong face masks wearing, and no face mask-wearing. In the face identification phase, the suggested technique obtained 98.7 percent recognition rate. By categorising seven face expressions with VGG-16, the authors published a Deep learning real-time facial emotion classification and identification approach in [18]. The proposed model was trained on the KDEF dataset and attained an 88% identification rate.

The above-mentioned analysis represents the views of several scholars on mask detection, categorisation, and various approaches used to combat it. This classification model has been named DTLMV2 and was formulated with the help of OpenCV and TensorFlow deep neural network frameworks, containing a Single Shot Multibox Indicator object identification model. A thorough explanation of the suggested approach and its analysis is discussed in Sections 3 and 4.

3. Materials and methods

This section describes the proposed methodology of real-time deep learning mask classifier DTLMV2 is as follows. The first step in classifying whether a person in a crowd has worn a mask appropriately is that the algorithm learns via a suitable dataset. The dataset was explored in Section 3.1 for further information. Upon successful completion of training of classifier, an accurate face recognition model must be used to recognise the faces (masked or unmasked) results in, DTLMV2 model can determine whether the one is wearing a mask or not. The purpose of this study is to enhance mask detection accuracy while utilising as few resources as possible. This assignment was completed using the DNN framework from OpenCV, which includes a ‘Single Shot Multibox Detector (SSD)’ [19] object detection model and ResNet-10 [20] as its backbone architecture and SSD object detection

M. Gupta, G. Chaudhary, D. Bansal et al.  Applied Soft Computing 127 (2022) 109313
photos in this dataset were of symmetrical nature (due to the fact that most of the datasets were either biased for a particular race, which does not truly reflect reality, or the dataset was comprised of the similar type of augmented images and incorrect labels. As a result, building the correct dataset was the most effective for this proposed model, although it took a long time. The dataset used to validate the algorithm in a particular way was a hybrid of open-source datasets and images, including augmenting datasets from Kaggle’s [22] and randomising it.

Used dataset comprising images of individuals wearing facial masks or people travelling in the metro and other public spaces as shown in Figs. 1 and 2. Some of the pictures have been used in the dataset which have been created by the methodology of a dataset created by a user named “Prajna Bhandary” to take typical photos of people’s faces and add face region. Face features such as eyebrows, eyes, mouth, nose, and jawline using face region have been detected easily. This created a dataset in an arbitrary method by using a mask on a non-masked human photograph. The dataset includes 3771 images, having the label “masked” and “unmasked” having 3743 images to result in well trained or learned dataset. The dataset has been made available [23–25].

3.1. Datasets

There is a plethora of datasets available for overcrowded public spaces. Thanks to different movements and protests across the world like the Black Lives Matter of 2020, where 15–20 million people joined. Created dataset for this work differs from the rest of the architecture in the sense that it included pictures of different kinds of races of people like Caucasians, dark-skinned, Asians and so on. The culmination and analysis of datasets was done solely due to the fact that most of the datasets were either biased for a particular race, which does not truly reflect reality, or the dataset was comprised of the similar type of augmented images and incorrect labels. As a result, building the correct dataset was the most effective for this proposed model, although it took a long time. The dataset used to validate the algorithm in a particular way was a hybrid of open-source datasets and images, including augmenting datasets from Kaggle’s [22] and randomising it.

Used dataset comprising images of individuals wearing facial masks or people travelling in the metro and other public spaces as shown in Figs. 1 and 2. Some of the pictures have been used in the dataset which have been created by the methodology of a dataset created by a user named “Prajna Bhandary” to take typical photos of people’s faces and add face region. Face features such as eyebrows, eyes, mouth, nose, and jawline using face region have been detected easily. This created a dataset in an arbitrary method by using a mask on a non-masked human photograph. The dataset includes 3771 images, having the label “masked” and “unmasked” having 3743 images to result in well trained or learned dataset. The dataset has been made available [23–25].

3.2. Pre-processing

The information about masked face detection and implementation had wrongly augmentation images, a lot of noise, and the photos in this dataset were of symmetrical nature (due to the majority of an abundance of Asian faces). The data from the above-mentioned datasets were resized into 224*224. They were then analysed and then further converted to a NumPy array, as good dataset represents eventually the kind of accuracy of the prototype to be achieved on it. This section covers how to prepare data after pre-processing and then train on it. Data cleaning was performed on the datasets, wherein the corrupt images which are not clear were removed and then the whole dataset was randomised to remix the images that looked similar consecutively. To sort the data in lexicographical order, firstly a method has been constructed that is sorted in an alphanumeric manner. Pre-processing begins with a function that takes a folder as input to the dataset, imports all of the pictures from the folder, and uses the DTLMV2 model to reformat the images.

After the data has finished being sorted alphabetically, the images are further reduced into multidimensional arrays known as tensors in TensorFlow. These tensors help to perform mathematical operations, matrix and element-wise multiplication. In TensorFlow, each data input is mostly always represented by a tensor, and is frequently a vector. The output (or value to be predicted) under supervised training is also a tensor. To increase the efficiency, the list is converted into a NumPy array. After that, data augmentation is used to increment the number of pictures in the dataset. This further improves the model’s accuracy after it has been trained. Further label binarizer is used as “one-hot encoding” for applying fit-transform to labels. A label Binarizer class in SciKit Learn takes categorical data as input and outputs a NumPy array. Preprocessing categorical features for machine learning models using one hot encoding is a frequent practice.

This form of encoding creates a new binary feature for each potential category and assigns a value of 1 to each sample’s feature that matches to its original category. If your data just includes two types of labels, which can be taken as input directly into a binary classifier. As a result, one column is sufficient to capture two classes in a One-Vs-Rest manner. A column vector is created by converting binary targets to a column vector.

3.3. Data augmentation

Since there is lack of an appropriately distributed data for the suggested model to train, an immense volume of data is required for effective training of the DTLMV2 model. This problem is solved using the data augmentation approach. Methods having parameters like rotating (rotation_range = 20), zooming (zoom_range = 0.15), shifting (width_shift_range = 0.2, height_shift_range = 0.2), shearing (shear_range = 0.15), and flipping (horizontal_flip = True) the image are employed in this way to create several variations of a similar photos. To set the pixels in the input region, the Fill mode option has been set to “nearest”. For the data augmentation procedure in the suggested approach, picture augmentation is employed. A function for the purpose of data creation is built for augmenting images, and it outputs test and trains chunks of data.

3.4. Deep transfer learning on MobileNetV2

MobileNetV2 is a Multilayer Neural System that was developed to handle the classification. ImageNet pretrained weights were imported using TensorFlow. The fundamental layers are then stored to minimise the impairment from previously learned properties. Then, further trainable layers are stacked and trained on the collected dataset to uncover the properties that differentiate a mask-wearing from a no mask. After that, the model is quite well, and the values are saved. Prior-learned algorithm save time & expense by allowing one to employ existing biased weights while preserving previously learned features.
3.4.1. Layers of MobileNetV2

- **Convolutional Layer**
  The Deep Convolution Network’s basic core element is this level. Convolution is a scientific word that indicates the combination of mathematical functions to produce a output function. It employs a sliding window technique to aid in the extraction of characteristics from a picture. This assists in the creation of feature maps.

- **Pooling Layer**
  The use of pooling techniques can speed up calculations by reducing the size input size matrix without sacrificing many properties. There are several types of pooling procedures that may be used, some of which are described below:
  - **Max Pooling:** It uses the highest value in the designated area in which the kernel is now located as the values for the matrix value created for that cell.
  - **Average Pooling:** It represents the mean of all values present in the area in which the kernel is located and uses that value as the matrix value for that cell’s matrix value.

- **Dropout Layer**
  Removing the prototype’s random biased neurons, helps to prevent overfitting that can occur during training. These neurons can be found in both hidden and visible levels. The dropout ratio may be modified to alter the chance of a neuron being dropped.

- **Non-Linear Layer**
  In a fully convolutional, a non-linearity layer is composed of an activation method takes the extracted features created by the previous layer and generates the result in activation map. Because the activation method is a component action on the input volume, the proportions of the outputs and inputs are the same.

- **Fully-Connected Layer**
  A neural network’s convolution layer is essentially a recurrent neural network (usually a 2- or 3-layer MLP) that seeks to translate the activation volume of previous various layers into a class probability distribution. The purpose of the fully connected structure is to tweak the weight parameters to construct a stochastic probabilistic model that shows that each class rely on the activation maps created by convolutional, rectification, non-linearity, and pooling layers. Except for the input layer, individual completely linked layers behave identically like the layers of a multilayer perceptron.

- **LeakyRELU**
  Here the ReLU activation function has specified as a very tiny linear component of x rather than as 0 for negative values of inputs(x). This activation function’s formula is as follows.

\[
 f(x) = \max(0.01 \times x, x) \quad (1)
\]

If it receives a positive input, it returns x; if it receives a negative input, it returns a very small value equal to 0.01 times x. As a result, it also produces an output for negative values. By making this minor change, the gradient on the left side of the graph becomes non-zero.

- **Softmax activation function**
  In general there are not even one final figures produced by the neural network. However, it is essential to decrease these values to integers from zero to one, which indicate each class’ probability. The Softmax function plays this job.

\[
 \sigma: R^k \rightarrow (0, 1)^k \quad (2)
\]

\[
 \sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad \text{for} \ j = 1, \ldots, K \quad (3)
\]

- **Linear Bottlenecks**
  Since many matrix products cannot be limited to a particular mathematical function, non-linear methods were employed in neural networks. Concurrently, the activation ReLU function, which is often employed in neurons, eliminates values less than 0. This uncertainty can be mitigated by growing the number of channels in order to boost the network’s capacity. For squeezing the layer upon layer where omit ties are coupled, the converse with flipped residual blocks must be performed. This has a negative impact on network performance. The authors proposed a linear bottleneck, in which the final iteration of a residual block has a linear output before it is added to the starting activations. Detailed analysis for the proposed model’s workflow is shown below in Fig. 3.

3.5. Proposed algorithm

The proposed DTLMV2 as shown in Fig. 4 technique has been presented in detail using the algorithm given below. The visuals were first processed and learned the entire dataset. Furthermore, the model that was developed in the first section was utilised to recognise the face mask with the required accuracy rate. Initially, images were utilised as input, with their pixel values being scaled, resized, normalised, and turned into a single dimension array. To increase the accuracy of the photographs, they were then submitted to a data pre-processing method. The data was then split into learning and evaluating batches and the MobileNetv2 model was employed. To assemble the whole model, “adam” optimiser was utilised.

SGD optimiser with a learning rate of 1e-2 has been set up. SGD is a simplified variation of classical gradient descent in which the stochasticity comes from computing the gradient at each descent using a random subset of the data (mini-batch).

In stochastic circumstances, SGD is classified as a non-convex optimisation issue, since researchers only observe a subset of the data at any given time. SGD is a method of generalisation beyond the training set that is an extension of the gradient descent algorithm. The term ‘stochastic’ refers to a system or process that has a random probability associated with it. Each iteration has its own data collection. The term “batch” is used in Gradient Descent to refer to the total number of samples from a dataset that are utilised to calculate the gradient for each iteration. The decay is then set to equal the learning rate divided by the total number of epochs over which the network is being trained. Keras uses the following learning rate schedule internally to modify the learning rate after each batch update. Finally, the MobileNetv2 model has been loaded from Keras. The training batches are trained on this loaded model using ImageNet weights and compiled using adam optimiser, which has a parameter for linear learning rate decay and scheduler.

\[
 lr = \frac{\text{init}\_lr}{1.0 + \text{decay} \times \text{iterations}} \quad (4)
\]

Furthermore, this trained in the preceding section was utilised on both static picture classification and real-time video streaming classification. When the human faces are recognised using SSD, the result shows a bounding box including the facial features of an individual wearing a mask.

4. Experimental results

The suggested model was assisted via several metrics such as precision, recall, F1 score, accuracy, sensitivity, and specificity, as stated in Eqs. (5)-(8). NVIDIA TESLA P100 GPUs have been used here. All the analysis were carried out on Kaggle GPU VM with a TESLA P100 GPUs processor (1,32 GHz), 13 GB of RAM and 16 GB
of VRAM. In this study, the Kaggle Notebook with Python 3.6 kernel was chosen for the formation and execution of the several experimental trials.

\[
\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}
\]  \hspace{1cm} (5)

\[
\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}
\]  \hspace{1cm} (6)

\[
\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]  \hspace{1cm} (7)

\[
\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{False Negative} + \text{True Negative} + \text{False Positive}}
\]  \hspace{1cm} (8)

True positive results in the calculations above relate to photos that were classified true and yielded true results after being predicted by the model. True negative, on contrary, refers to photos that were classified as false and also resulted in false in actual assessment. False-positive photos are those that were classified true but turned out to be wrong after evaluation. False-negative photos are those that were classified false but turned out to be correct after prediction. In Fig. 5, a form for contrasting labels, simulated results, and the actual labels that the algorithm was supposed to predict is shown. It illustrates the point at which the model becomes bewildered. To visualise 2D matrix data in form of graph, the confusion matrix is generated using a heatmap. It has found 991 true positives and 1000 false negatives. Out of 1551 photos used for validation, there were 18 false-positives, 20 false-negatives, and 776 genuine negatives.

Since the classes were evenly distributed, accuracy was an excellent starting point. Precision was used to determine the proportion of positive predicted values. The recall score offered a classifier the capacity to discover all positive samples, whereas the f1 score was a measure of test accuracy. These standards of assessing the efficacy of the model were selected as they can produce the efficient results in a well-balanced dataset. In Table 1, the different metrics of classification report for masked and unmasked dataset have been illustrated. Accuracy comparison and F1 score between different models has been depicted in Table 2.

4.1. Evaluation — DTLMV2 model

To diminish the binary classification problem, a deep transfer learning-based architecture is used to resolve the problem in this study. In this study, Keras, an enhanced artificial neural networks API, is implemented to create a classifier. The data sets are
Fig. 5. Shows 2-class Confusion Matrix.

Table 1
Classification report.

|             | Precision | Recall | F1-score | Support |
|-------------|-----------|--------|----------|---------|
| Masked      | 0.98      | 0.95   | 0.97     | 776     |
| Unmasked    | 0.95      | 0.98   | 0.97     | 775     |
| Accuracy    |           |        | 0.97     | 1551    |
| Macro average| 0.97    | 0.97   | 0.97     | 1551    |
| Weighted average | 0.97 | 0.97   | 0.97     | 1551    |

Table 2
Comparison of accuracy and F1 score between different models.

| Architectures used | Year | Accuracy (%) | F1 Score |
|--------------------|------|--------------|----------|
| LeNet-5            | 1998 | 84.6         | 0.85     |
| AlexNet            | 2012 | 89.2         | 0.88     |
| SSDMNv2            | 2020 | 92.64        | 0.93     |
| DTLMV2 (proposed model) | 2022 | 97.04      | 0.97     |

Fig. 6a. Accuracy curve.

Fig. 6b. Loss curve.

Fig. 7a. Input image.

Fig. 7b. Classified output image.

The accuracy, region that under classification report, confusion matrix, receiver operating characteristic curve (ROC), and model comparative were the assessment measures employed in this work. The charts are based on prototype accuracy, stimulates using matplotlib and MATLAB. The accuracy indicates the percentage of correctly predicted masked individuals by the machine using the specified model. Figs. 6a and 6b depicts the classic characteristics of step-based learning rate scheduling: Reduced training/validation loss. As the rate of learning slows down, there is a significant improvement in training/validation accuracy. This is most noticeable in the first two decreases (epochs 15 and 30), after which the declines become less significant. This sort of sharp decline is a classic indication of the use of a step-based learning rate schedule.
4.2. visualisation of results

Face mask predictions are provided by DTLMV2 utilising MobileNetV2 on sample (i.e., 5) test photos. The correct means of covering the face of a person with mask is depicted by the rectangular bounding box, having an accuracy score on the upper left, whereas the completely wrong way to covering a face with mask is depicted by the rectangular red bounding box. The final model is trained from the train dataset’s sequence and classifications to produce predictions as shown from Figure 7 to Figure 11 as Figs. 7a, 8a, 9a, 10a and 11a shows input image whereas Figs. 7b, 8b, 9b, 10b and 11b shows classified output images in a crowd. The face recognition model obtained 97.01% accuracy
on validation data, 98% accuracy on training data and 97.45% accuracy on testing data as mentioned in Table 3.

5. Conclusion and future scope

This study proposes and implements a face mask detector that can recognise whether the face of a person is covered with a proper mask at places with mandatory regulations of COVID-19.

| Dataset distribution (photos) | Accuracy (in %) | Loss  |
|-------------------------------|----------------|-------|
| Validation                    | 1551           | 97.01 | 0.0593 |
| Testing                       | 1501           | 97.45 | 0.0536 |

A two-stage detector structure was employed in this model. On a dataset of 7514 images, the model was trained using transmission learning with the MobileNetV2 framework. It employed a Haar cascade-based feature extractor and methodology of classification to extract the region of interest (ROI), resulting in identifying the sections and finally passed them on to the learning classification method to acquire the preferred outcome. Training accuracy was 98 percent, while testing accuracy was 97.45 percent, according to the findings of the experiments. When the model was employed in real-time video streaming to determine whether the facial area of a certain group of individuals were having any mask or not, it produced impressive results. The suggested and constructed face mask detector in this work may be utilised as a system of surveying crowds in universities, malls and departmental stores to check that group of individuals are wearing face masks to avoid the spread of disease. In further scope of improvement, many outlier data point and missing data shall be considered using feature engineering algorithms. These may also include some form of ensemble tuning and cross validation. For future study, one-stage detector frameworks such as Single Shot Detectors and algorithms like YOLO can be employed for detection of facial masks, reducing computing time and therefore making detection faster. Additionally, alternative datasets and dynamic framework-based solutions may be utilised to train and assess the face detector model to improve its performance in real-time photos and videos. Scarves and fabric, for example, are frequently used as masks. The model recognises it as a face mask since it has the same characteristics as a face mask. The model would be more practical considering several use-cases of real world if the features were selected more precisely. With regards to further enhancing research prospects, one can utilise the dataset supplied in this study for more sophisticated models; the DTLMV2 model should confidently aid the respectful and responsible authorities in these times of crisis and pandemic that has wreaked havoc in almost all of the globe.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
References

[1] World Health Organization, Severe acute respiratory syndrome (SARS), 2020, https://www.who.int/csr/sars/en/. (Accessed 5 October 2020).

[2] COVID-19 dashboard by the center for systems science and engineering (CSSE) at Johns Hopkins University (JHU), 2020, ArcGIS. Johns Hopkins University. Retrieved 8 https://reliefweb.int/report/world/coronavirus-covid-19-global-cases-johns-hopkins-csse?gclid=CjwKCAjwve2TBByEiwAaMIF5y3CQw4VbTvLXIf9xqXmMqIqNfdgve2Z55E2-prYwq_L0AOklyBoGfB6QAvD_BwE.

[3] S. Taneja, A. Nayyar, P. Nagrath, Face mask detection using deep learning during COVID-19, in: Proceedings of Second International Conference on Computing, Communications, and Cyber-Security, Springer, Singapore, 2021, pp. 39–51.

[4] World Health Organization (WHO), Coronavirus disease (COVID-19) pandemic, 2020, https://www.who.int/emergencies/diseases/novel-coronavirus-2019. (Accessed 14 October 2020).

[5] C. Li, R. Wang, J. Li, L. Fei, Face detection based on YOLOv3, in: Recent Trends in Intelligent Computing, Communication and Devices, Springer, Singapore, 2020, pp. 277–284.

[6] P. Viola, M. Jones, Rapid object detection using a boosted cascade of simple features, in: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit, Vol. 1, CVPR, Feb. 2001, pp. 1–511.

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

[8] J.S. Park, Y.H. Oh, S.C. Ahn, S.W. Lee, Glasses removal from facial image using recursive error compensation, IEEE Trans. Pattern Anal. Mach. Intell. 27 (5) (2005) 805–811.

[9] W. Yang, Z. Jachun, Real-time face detection based on YOLO, in: 2018 1st IEEE International Conference on Knowledge Innovation and Invention, ICKII, July, IEEE, 2018, pp. 221–224.

[10] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems, 2015, p. 28.

[11] H. Li, Z. Lin, X. Shen, J. Brandt, G. Hua, A convolutional neural network cascade for face detection, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 5325–5334, http://dx.doi.org/10.1109/CVPR.2015.7291770.

[12] M. Loey, G. Manogaran, M.H.N. Taha, N.E.M. Khalifa, A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic, Measurement 167 (2021) 108288.

[13] J. Zhang, F. Han, Y. Chun, W. Chen, A novel detection framework about conditions of wearing face mask for helping control the spread of COVID-19, IEEE Access 9 (2021) 42975–42984, http://dx.doi.org/10.1109/ACCESS.2021.3066538.

[14] B. Wang, Y. Zhao, C.L.P. Chen, Hybrid transfer learning and broad learning system for wearing mask detection in the COVID-19 era, IEEE Trans. Instrum. Meas. 70 (2021) 5009612, http://dx.doi.org/10.1109/TIM.2021.3069844, 1–12.

[15] S. Taneja, A. Nayyar, P. Nagrath, Face mask detection using deep learning during COVID-19, in: Proceedings of Second International Conference on Computing, Communications, and Cyber-Security, Springer, Singapore, 2021, pp. 39–51.

[16] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, J. Hemanth, Sdnnv2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2, 2021 Sustain. Cities Soc. 66 (2021) 102692.

[17] B. Qin, D. Li, Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19, Sensors 20 (18) (2020) 5236.

[18] S.A. Hussain, A.S.A.A. Balushi, A real time face emotion classification and recognition using deep learning model, J. Phys.: Conf. Ser. 1432 (2020) 012087, http://dx.doi.org/10.1088/1742-6596/1432/1/012087.

[19] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg, Sd: Single shot multibox detector, in: European Conference on Computer Vision, Springer, Cham, 2016, pp. 21–37.

[20] D. Anisimov, T. Khanova, Towards lightweight convolutional neural networks for object detection, in: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS, 2017, pp. 1–8, http://dx.doi.org/10.1109/AVSS.2017.8078500.

[21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, MobileNetV2: Inverted residuals and linear bottlenecks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520, http://dx.doi.org/10.1109/CVPR.2018.00474.

[22] Sahana Srinivasan, R. Rujula Singh, Ruchita R. Biradar, S. Revathi, COVID-19 monitoring system using social distancing and face mask detection on surveillance video datasets, in: 2021 International Conference on Emerging Smart Computing and Informatics, ESCI, IEEE, 2021, pp. 449–455, http://dx.doi.org/10.1109/ESCI50559.2021.9396783.

[23] Larxel, Face Mask Detection, Available at: https://www.kaggle.com/andrewmvd/face-mask-detection.

[24] MaskDataset5, Available at: https://cutt.ly/vOlE67t.

[25] P. Bhandary, Mask Classifier Observations, Available at: https://github.com/prajnasb/observations.