Face mask recognition using CNN

Mansata Dhairya
Computer Science Engineeriing
Nirma University
Gujarat, India
20bce154@nirmauni.ac.in

Abstract—There are numerous different viruses in the coronavirus family. Some of these can spread the common cold to people. It spreads from person to person and is actually quite contagious. Therefore, it has become crucial that we safeguard both ourselves and those who are close to us from this circumstance. Maintaining a safe distance from others, washing our hands every two hours, using hand sanitizer, and—most importantly—wearing a mask Face masks are frequently used in public in China and other countries since the latest coronavirus disease pandemic started. The Health Centre's recommendation states that recent research have shown that many coronavirus patients are asymptomatic (or "asymptomatic") and that others are pre-symptomatic (or "pre-symptomatic"), or able to pass on the corona virus to other people before even developing symptoms. This would mean that the virus can spread among people who are in close proximity and also by coughing, or sneezing or speaking, even if they are asymptomatic.

Index Terms—component, formatting, style, styling, insert

I. Introduction

There are numerous different viruses in the coronavirus family. Some of these can spread the common cold to people. It spreads from person to person and is actually quite contagious. Therefore, it has become crucial that we safeguard both ourselves and those who are close to us from this circumstance. Maintaining a safe distance from others, washing our hands every two hours, using hand sanitizer, and—most importantly—wearing a mask Face masks are frequently used in public in China and other countries since the latest coronavirus disease pandemic started.

A. Motivation

One survey says 90% Indians are aware, but only 44 of them wearing a mask; discomfort is a key justification for not complying. In light of the current situation, we thus require the assistance of cutting-edge technology to detect masks and prevent people without masks from entering any "No Entry Without Mask" zones in order to enforce the mask wearing standards. Face identification is a straightforward process that uses technology, particularly hardware like video cameras, to recognise faces and facial traits. By comparing it to a database of recognised faces, the face recognition app or programme employs biometrics to map the facial features from any image or video. The use of facial recognition software will grow as a result of government initiatives and rising need for surveillance systems to improve security. Scientists employed by the US Department of Defense conducted the first documented attempts to develop a facial recognition programme in the 1960s. These early applications used computers that could recognise people from images, but they lacked the capability to analyse real-time video. One of the earliest systems capable of evaluating real-time audio and video and tracking people's faces within it was created in 1984 by Carnegie Mellon University. The CMU system had potential uses in security in addition to its original goal of assisting blind persons in recognising members of their social networks. Facial recognition systems that can recognise people in real time and from recorded video did not become available until the late 1990s. FaceCom created one of the first face recognition algorithms that were commercially available in 2001. This technique was scalable and has received permission from various nations to be used in biometric passports. The ihai was promoted as a security camera that could recognise people using face recognition technology when Hewlett-Packard started marketing it in 2003. The only way that this product could recognise persons was through old pictures or movies; it had no real-time capabilities. In addition, it was not apparent how ihai arrived at its claimed capacity of over one billion facial recognitions or whether these figures were reliable. FaceFirst released FaceFind in 2006 as a comparable tool. With the use of server-based software, this system could identify people in real time, but at the time it lacked any means of recording data or video.

B. Contribution

Face mask detection has seen a rise in demand as a result of the global economic crisis. It is one such piece of technology that can identify a person by acknowledging their face through a mask. It employs an individual's biometric data along with an AI-based pattern identification systems. It isolates facetraits and divides them into various groups. Additionally, it may acknowledge individuals who are not wearing masks by producing an alarm or notification to alert security or authority. Through software, a smartphone app, a device, or a website, they can determine who has not worn masks to disguise their faces.

C. Problem Statement

Recently, face coverings have become law in more than fifty countries worldwide. Markets, public transportation, offices,

and retail establishments require that people conceal their faces visibly. Retail businesses frequentlyuseprogrammingto track the number of customers entering their locations.They could also desire to evaluate the effects of digital presentations and transient screens.

II. DATASETS SELECTION

- 7553 colour photos are available in both folders —one with mask and the other without mask the data set.
- Label with mask and without mask are used to identify images. 3828 photos of faces are withoutmasksand3725 images of faces are with masks.
- Our dataset has to be divided into two parts: a training dataset and a test dataset. We would need a model that would work well on a dataset that is not seen before(test data).
- The actual subset of the dataset we use to build the model is called the Training set. The model notices and learns from this data and then refines its limits and boundaries.

Figure 1: Dataset

III. HOW DOES NEURAL NETWORK WORK?

Neural networks, often known as synthetic or artificial neural networks, serve as the foundation for deep learning techniques. They imitate the manner in which actual neurons connect with one another by taking their cues from how the human brain works.

An input layer, perhaps one or more hidden layers, andan output layer are all present in every neural network. The threshold value and weight of every neuron are related to those of the others. If a neuron's output exceeds thethreshold value defined for that node, it is activated and starts relaying data to the following layer of the network. The next tier ofthe network will not get any data if this is not the case.Forneural networks to grow and improve their accuracy over time, training data is necessary. However, if these learning algorithms are optimised for accuracy, they can be useful machine learning and artificial intelligence toolsthathelpus classify and cluster data more effectively. Speech or image recognition can be finished in minutes ratherthanhours when compared to expert human identification.

Each node should be given its own linear regressionmodel, complete with input data, weights applied, a thresholdvalue, and an output. The calculation can look likethisasanexample:

$$y = \sum_{i=0}^{n} w^i x^i + b \quad (1)$$

Following the identification of the input layer, weights are assigned. Each variable's significance is determinedbythese weights, with larger weights having a greater impact on the outcome than smaller ones. Each input is then multipliedby the appropriate weight before being joined together. An activation function will first determine the output, and then the output will be determined by that function. The neuron "lights" (activates), sending information to the network'snext tier if the output rises above a predetermined threshold.As a result, the node that comes after one neuron's output becomes that node's input. A feedforward network is one in which information is sent from one layer to the next.

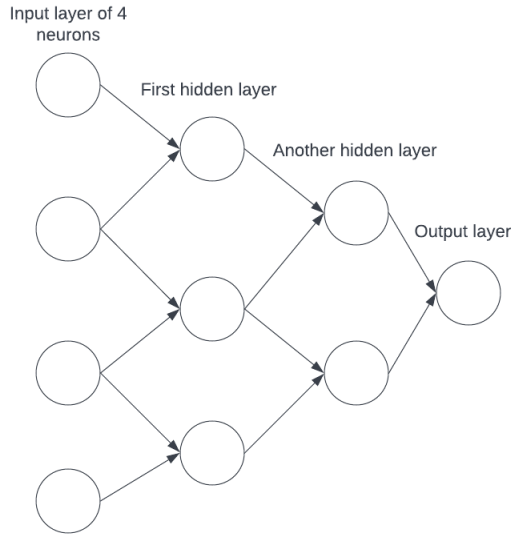


Figure 2: Sample Artificial Neural Network

In order to train the algorithm, supervised learning, or labelled datasets, will be used as we start to thinkaboutmore practical applications for neural networks, includingimage recognition or classification. As we start to trainthemodel,we will want to evaluate its precision using a loss function(cost function). Our ultimate objective is to minimise our cost function in order to increase the accuracy of our network.The model uses reinforcement learning and the cost function to modify its weights and bias in order to arrive at the point of convergence, also referred to as the local minimum. Our

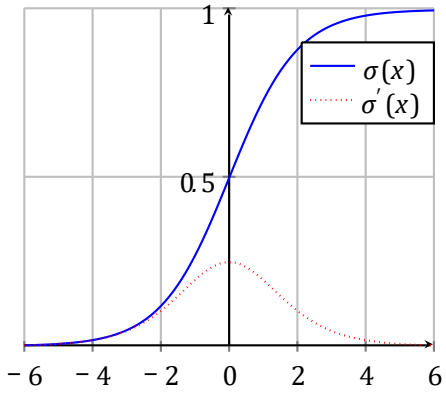
model is able to select the best course of action to minimise errors thanks to the technique's use of gradient descent to update its weights.

The majority of deep neural networks are feedforward, which means that data only flows in one direction from input to output. Our model can also be trained via backpropagation, which includes going from input to output in the opposite direction. By effectively changing and fitting the model(s)' parameters, backpropagation enables us to measure and manage the error related to each neuron.

In this study, feedforward neural networks have received the majority of our attention. An input layer, a hidden layer, and an output layer make them up. Since the bulk of problems encountered in the actual world are nonlinear, these neural networks use sigmoid neurons instead of perceptrons. Neural networks utilised in computer vision, natural language processing, and other fields are built on top of these models. Data is regularly inputted into them.

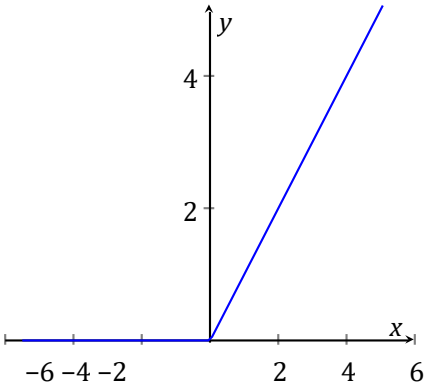
CNNs are used for image recognition, pattern recognition, and computer vision. These networks use matrix multiplication and the concepts of linear algebra to find patterns in an image. We will now train a basic artificial neural network for picture categorization in order to achieve our goals. Our photos have been downsized to the shape (96,96,3), resulting in an input array of 96X96X3 (3 is for RGB channel), which will result in a flattened one-dimensional array of 27648 neurons for our input layer. With neurons numbering 3000 and 1000, respectively, we have utilised two densed network hidden layers in between, each of which has relu as an activation function. Two neurons make up each of our output layers, each with the sigmoid activation function and a threshold value of 0.5 for both classes. A neuron will light up if it receives a value larger than 0.5, signifying whether the given image belongs to the specified class or not.

- Sigmoid: Here, the activation function is a mathematical function called the sigmoid. It typically has a distinctive S-shaped curve. There are numerous sigmoid functions, and the logistic function is the one employed in this case. It essentially converts the real value of the neuron into a probability by mapping our complete number line to the range of 0 to 1.



$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

- Relu: Another activation function, relu, is used in these concealed levels. If the input is positive, the rectified linear function will output the value immediately. If not, it would be 0. The activation function that is most frequently used in ANNs, CNNs, and multi-layered perceptrons.



$$f(x) = max(0,x) \tag{3}$$

IV. CNN

The deep learning algorithm known as Convolutional Neural Networks (CNN) is well renowned for its ability to classify images. It deals with knowledge representation and has a higher degree of adaptability for analysing visual images. As opposed to the simple ANN methodology, the given methodology divides each image into various objects and visualises each one separately, allowing for a more accurate evaluation of each layer.

V. HOW CNN WORKS ON AN IMAGE?

It moves a straightforward grid over the entire image, compares the current covered image by the grid by matching each pixel inside the covered region, and then calculates the likelihood that an item or pattern (in our example, a face mask) is covered in the current part of the image. Therefore, rather than being static, the grid scans the entire image in search of a specific pattern before generating the probability.

We utilised the adam optimizer and the sparse categorical cross entropy loss function (cost function). Adam is a technique for optimization that makes use of the exponential decaying average of previous gradients. Logarithmic loss, logloss, and logistic loss are other names for cross-entropy loss. Each projected class probability is compared to the actual class probability, and a loss is produced that penalises the probability according to how far it deviates from the real expected value. The fine has a logarithmic scale. It produces a large score for differences approaching 1 and a small score for differences approaching 0. A cross-entropy loss of zero indicates a perfect model.

VI. Data Splitting

We have around 7553 images in our dataset of which We used the inbuilt train test split from sklearn which divide our dataset randomly in train and test data sets. We kept our test data size to be 15% which means we have 1133 images in test data and the other images in our train data set. We split our train data set into train and validation data set(15% of train_ data set i.e. 963 images). Our main focus is to get better precision and accuracy for our test data.

VII. Implementation of ANN

On our training dataset, we achieved an accuracy of 93% accuracy using Simple ANN, while on our test dataset, we acquired an accuracy of 89%. The network is trained using the 20 epochs and the aforementioned parameters. We do not anticipate it to perform well for a problem resembling image classification because it was just a basic neural network with a couple of layers. The ANN classification report is as follows:

```
Accuracy on test data is:  0.8980807662010193

Classification Report:
              precision    recall  f1-score   support

           0       0.88      0.92      0.90       761
           1       0.92      0.87      0.89       750

    accuracy                           0.90      1511
   macro avg       0.90      0.90      0.90      1511
weighted avg       0.90      0.90      0.90      1511
```

Figure 3: Classification report of ANN

Additionally, ANN will need to perform a lot more computations and is sensitive to an object's placement within an image. CNN, on the other hand, has the ability to both detect the same object at several locations within the same image and at different times inside the same image. CNN has a larger influence over picture classification as a result, as will be discussed in a later section.

VIII. Implementation of CNN

Here, a certain pattern or object is allocated to each tier in the network. The input, hidden, and output layers are identical to those of an ANN. The following list of additional parameters, including MaxPooling, Padding, Convolutional filters, kernel size, and many others, is available:

- Filters: The feature (objects or patterns) detectors are filters. It won't matter where an object appears in an image; the system will still detect it. Each filter has a kernel size of nxm, which designates a grid of that size that will extract features from the picture moving from top to bottom and then from left to right using various mathematical calculations. It will eliminate superfluous filters and concentrate on those that are required. This will act as if the image area were to be zoomed in ever increasingly at each filter to capture incredibly minute details. The probability that the object or filter we're looking for is there or nearby by nxm sizes will be displayed in a feature map as a result of this. Each of these filters has been combined with relu, resulting in an output function that is not linear.
- Pooling: By using pooling, we can decrease the amount of the grid of image that needs to be explored after every single convolutional layer, resulting in less computations. It reduces feature sampling and feature map dimensions by keeping the most crucial feature data. The size of the feature maps is reduced through pooling. If there are numerous convolutional layers with pooling between each layer, the subsequent layer will therefore have a smaller sample feature. To minimize the feature map, the next point is moved using a stride value (i.e we can take maximum value from each position in this feature map within p cross q or its average, or it can be in some other way). It also moves from top to bottom before moving from left to right in a feature map.

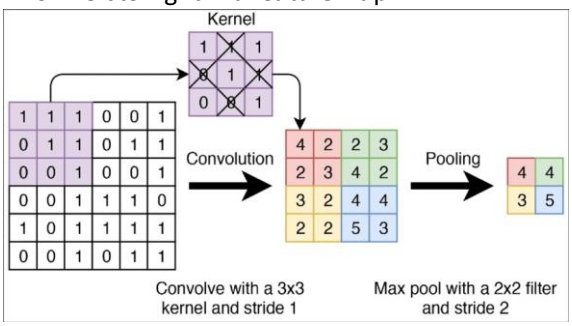


Figure 4: Convolutional network+pooling layer

CNN networks include convolution layers with pooling as one component. For feature extraction, it is employed. A pooling layer of stride 2x2 is employed after two convolution layers, one of filters 32 and one of filters 64, each with a 3x3 kernel size and activation function. For classification, feature maps that have been acquired after all convolution layers will be flattened into a single dimension array and supplied to fully connected densely fitted neural networks as an input after convolution layers. Then, the networks will function similarly to simple ANN networks.

Accuracy has clearly increased in both the train and test datasets. The accuracy for the test dataset was approximately 93%, whereas the accuracy for the training dataset was around 99%, demonstrating that CNN performed better for picture classification than ANN. Recall was about 96% and precision was around 91%, resulting in a higher f1-score of 94%.

```
epoch 18/20
189/189 [==============================] - 2s 10ms/step - loss: 0.0029 - accuracy: 0.9993
epoch 19/20
189/189 [==============================] - 2s 10ms/step - loss: 0.0359 - accuracy: 0.9891
epoch 20/20
189/189 [==============================] - 2s 10ms/step - loss: 0.0152 - accuracy: 0.9955
<keras.callbacks.History at 0x7f01ea811050>
```

Figure 5: Training accuracy(overfitting)

```
48/48 [==============================] - 0s 4ms/step
0.9338186631369954
              precision    recall  f1-score   support

           0       0.91      0.96      0.94       761
           1       0.96      0.91      0.93       750

    accuracy                           0.93      1511
   macro avg       0.94      0.93      0.93      1511
weighted avg       0.93      0.93      0.93      1511

[[730  31]
 [ 69 681]]
```

Figure 6: Classification report

However, 99% train accuracy and significantly lower test accuracy typically signify model overfitting. The phrase ”overfitting” in ML refers to a model that performs exceptionally well on a train dataset but poorly on a test dataset. Overfitting may occur for a variety of reasons, such as when the network tries to form a more complex curve to completely fit each of the train data or when the test dataset contains data of a different type that was not expected. The signal and noise will be separated using a successful machine learning system. The algorithm may end up ”memorizing the noise” rather than locating the signal if it is too flexible or complex (for example, if it has too many input features or is improperly regularized). This will lead to overfitting of model and this will lead to making predictions based on the noise. It will perform unusually well on i the training dataset available but it would give very poor result on new unseen data.

Data augmentation is one well-liked method for avoiding overfitting. A variety of methods collectively referred to as ”data augmentation” can be used to artificially increase the amount of data by producing more data points from current data.

This involves employing deep learning models to add new data points or make a few simple changes to the existing data. Data augmentation improves ML models’ performance and output by adding more unique examples to training datasets. We’ll try to make new images from the existing ones by cropping, flipping, zooming, and other image manipulations.

```
data_augmentation = keras.Sequential(
  [
    layers.experimental.preprocessing.RandomFlip("horizontal",
                                                 input_shape=(96,
                                                              96,
                                                              3)),
    layers.experimental.preprocessing.RandomRotation(0.1),
    layers.experimental.preprocessing.RandomZoom(0.1),
  ]
)
```

Figure 7: Data augmentation

One further convolutional layer with 32 filters and the same other parameters is added to prevent overfitting and to extract more significant information from training data rather than just memorising the noise. There is a dropout layer after convolution layer which randomly removes any number of neurons which will thus evert the ML model to memorise the noise. In a fully connected network, two concealed, dense layers of 64 and 32 are used. Our training accuracy dropped to 97.8% with the help of this network, but the test accuracy increased to almost 97%, which was ultimately our goal—to improve the performance of our model on test datasets.

Figure 8: Training accuracy of data augmented network

```
36/36 [==============================] - 0s 5ms/step
0.9646954986760812
              precision    recall  f1-score   support

           0       0.94      0.99      0.97       567
           1       0.99      0.94      0.96       566

    accuracy                           0.96      1133
   macro avg       0.97      0.96      0.96      1133
weighted avg       0.97      0.96      0.96      1133
```

Figure 9: Classification report of data augmented network

For the model to be evaluated, precision is a crucial metric since it shows us how accurately the model predicted the true values. It is the proportion of correctly classified positive samples (True Positives) to all of the positive samples that have been classified (either correctly or incorrectly). Also, precision of 97% is attained. Recall gauges how many actual positive labels the model properly identified. Recall of 96% is attained, bringing the f1-score—a metric that combines precision and recall—up to 97%.

IX. Results

In our model, we have used a dataset consisting of 7553 images of faces of people wearing mask or not wearing mask. We have implemented a convolutional sequential deep learning model with data augmentation in beginning, 3 convolution layers each followed by a pooling layer, 2 dense hidden layers and 1 dropout layers and a output layer of 2 neurons. The final activation function used is sigmoid so the output ranges between 0 and 1, if the output is below 0.5, it will lit the corresponding neuron. The accuracy of our model ranges between 97%. Figure 10. shows the variation in accuracy after each epoch. The test accuracy achieved is between 96%-97%.

Figure 10: Loss and Accuracy plot

A performance evaluating matrix of size NxNwhichiscalled

```
Epoch 29/30
189/189 [==============================] - 6s 31ms/step - loss: 0.0819 - accuracy: 0.9700
Epoch 30/30
189/189 [==============================] - 6s 31ms/step - loss: 0.0649 - accuracy: 0.9767
<keras.callbacks.History at 0x7fb787677e50>
```

```
cnn.evaluate(X_test,y_test)
```

```
48/48 [==============================] - 1s 8ms/step - loss: 0.0903 - accuracy: 0.9682
[0.09025644510984421, 0.9682329297065735]
```

Confusion matrix is used to evaluate the accuracy of our classification model. Here,target classes is denoted by N. It shows a tabular review of in total how many predictions a classifier made correct and incorrect. It is used to check a classification model's accuracy. It is also used to calculate different performance metrics of our ML classificationmodel such as F1-score, precision, recall and accuracy in order to check the effectiveness of our ML model. Following figure is the confusion matrix for the Convolutional Network webuilt, where the diagonal displays the accurate predictions forthe respective classes:

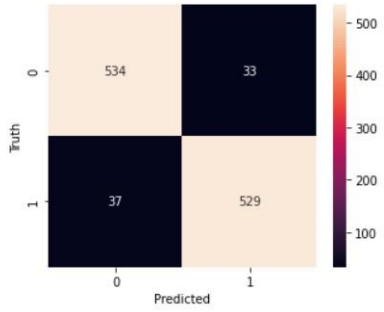


Figure 11: Confusion matrix

X. Evaluating our model with real images

XI. Applications

Face mask recognition has a variety ofpossibleapplications as facial recognition technology develops. Face mask recognition could, for instance, be used for security reasons, to deter criminality, or to identify people who must wear face masks in particular circumstances. Face mask recognition could also be utilisedinretailorcustomer service environments to make sure staff members are giving consumers the best service possible.Beforeitcanbe widely adopted, this technology presents a numberofcrucial concerns regarding security and privacy thatmustberesolved. Face mask recognition has a wide range of possibleuses,but there are still many unsolved concerns about the most effective ways to put this technology to work.

XII. Conclusion

Convolution Neural Network(A deep learningalgorithm)can very accurately classify images and help in imageprocessing. With multiple neural network layers, classificationofimages becomes better and more accurate. We have implemented CNN model to detect if the person is wearing a mask. Image processing is done with the help of deep learning techniques in medical sector. Our CNN model can prove to be very effective as it is quite accurate, input of an image of person wearing a mask is given and our model correctly predicts that he is wearing a mask

References

[1] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets:efficient convolutional neural networks for mobilevisionapplications," 2017, https://arxiv.org/abs/1704.04861.

[2] Wei Wang, Yutao Li, Ting Zou, Xin Wang, JieyuYou,Yanhong Luo, "A Novel Image Classification ApproachviaDense-MobileNet Models", Mobile InformationSystems, vol.2020,ArticleID 7602384, 8 pages, 2020. https://doi.org/10.1155/2020/7602384

[3] I. B. Venkateswarlu, J. Kakarla and S. Prakash, "Facemask detection using MobileNet and Global PoolingBlock,2020 IEEE 4th Conference on Information Communication Technology (CICT), 2020, pp. 1-5, doi: 10.1109/CICT51604.2020.9312083.

[4] M. S. Ejaz and M. R. Islam, "Masked Face Recognition Using Convolutional Neural Network," 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), 2019, pp. 1-6, doi: 10.1109/STI47673.2019.9068044

[5] Changjin Li, Jian Cao, and Xing Zhang. 2020. Robust Deep Learning Method to Detect Face Masks. In ¡i¿Proceedings of the 2nd International Conference on Artificial Intelligence and Advanced Manufacture¡/i¿ (¡i¿AIAM2020¡/i¿). Association for Computing Machinery, New York, NY, USA, 74–77. DOI:https://doi.org/10.1145/3421766.3421768