Indian Institute of Technology Indore

Discipline of Computer Science and Engineering

Minor Project in the course “Computational Intelligence”

Spring 2022-2023

Title: Real Time Face Mask Detection

Project Report

Team Members :-

Anushka Nile : 200001006

Sarvagya Kanungo : 200005040

Introduction :-

In this post pandemic world, Wearing face mask has become an essential aspect of our daily lives. However, enforcing the proper use of mask in public spaces has been challenging. Our solution here uses computer vision techniques to detect if a person is wearing a mask in real time.

Problem Statement :-

Real Time Face mask detection has become an important application in this post pandemic world. The main objective of the face mask detection problem is to accurately detect whether a person is wearing a face mask or not. It involves the usage of computer vision techniques to analyze real time video footage and provide alerts or notifications when a person is detected not wearing a face mask regardless the lighting conditions around the person or the type of face mask, he/she is wearing. It should be able to detect masks from all different angels or poses of the person.

Data Set :-

●We used Face-mask-detection-4k-images dataset for training and testing.

●It is a standard dataset which includes 4,000 images.

●It is equally divided into ‘Mask’ and ‘No Mask’ categories.

●The images are of various sizes, the faces are real as well as animated and the pictures are taken at different angles.

Model :-

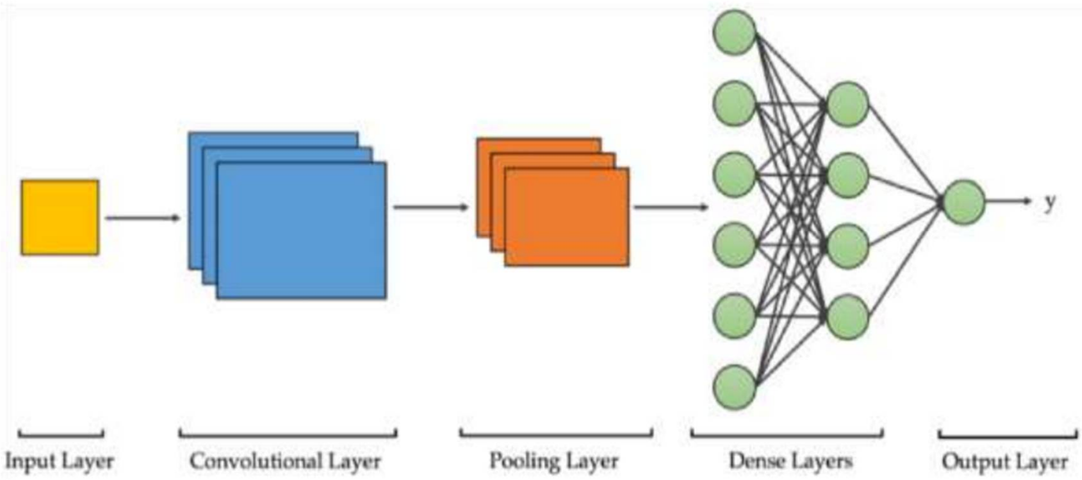
MobileNetV2 network is used and we add our own custom Fully Connected layer to it for perform binary classification on our dataset (masked and unmasked). MobileNetV2 is a lightweight architecture that is highly efficient in terms of memory usage and computational speed, making it well-suited for real-time applications and allows us to achieve high accuracy with minimal computational resources.

YOLO is primarily designed for object detection tasks in complex scenes, YOLO is a heavier and more complex model compared to MobileNetV2, and it requires more computational resources and training time.

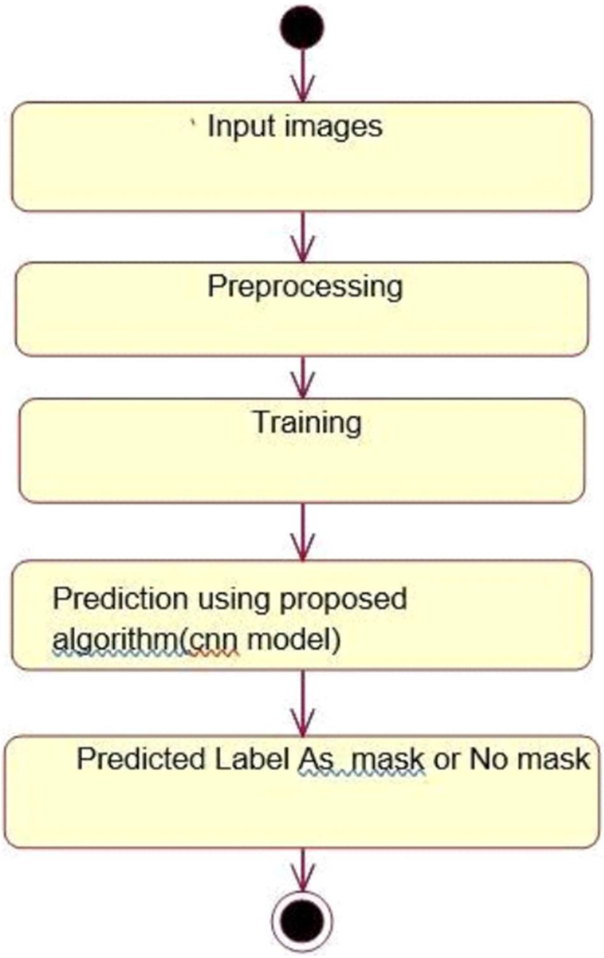
DenseNet is a good model architecture for many computer vision tasks, it may not be the best choice for real-time face mask detection due to its computational complexity and large number of parameters as compared to MobileNetV2.

We Design the head of the model that is to be placed on the top of the base model.

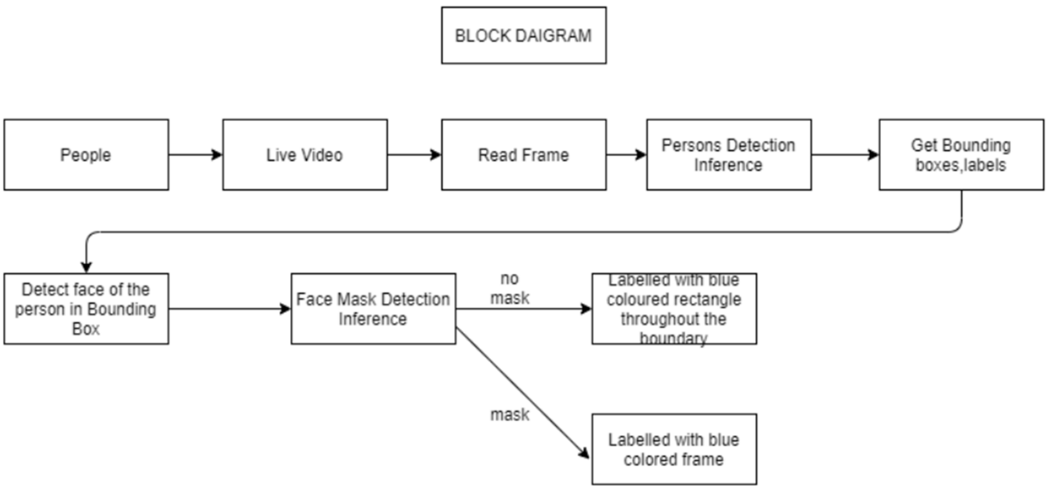
Layers:-



Activity Diagram:-



Block Diagram:-



Adam optimizer :-

Adam optimizer can adapt the learning rate for each parameter of the model based on the past gradient values and it also uses the moving average of the gradient instead of the gradient itself. This helps to achieve faster convergence and better accuracy, especially when dealing with large datasets or complex models.

ADAM combines the benefits of two other extensions of stochastic gradient descent (SGD), namely, Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp). it computes adaptive learning rates for each parameter and stores exponentially decaying average of past gradients and squared gradients.

Training and Testing :-

The model is trained on the head of the network using the fit() method of the model. The model is evaluated on the testing data using the predict() method and the classification report is printed to show the performance of the model.

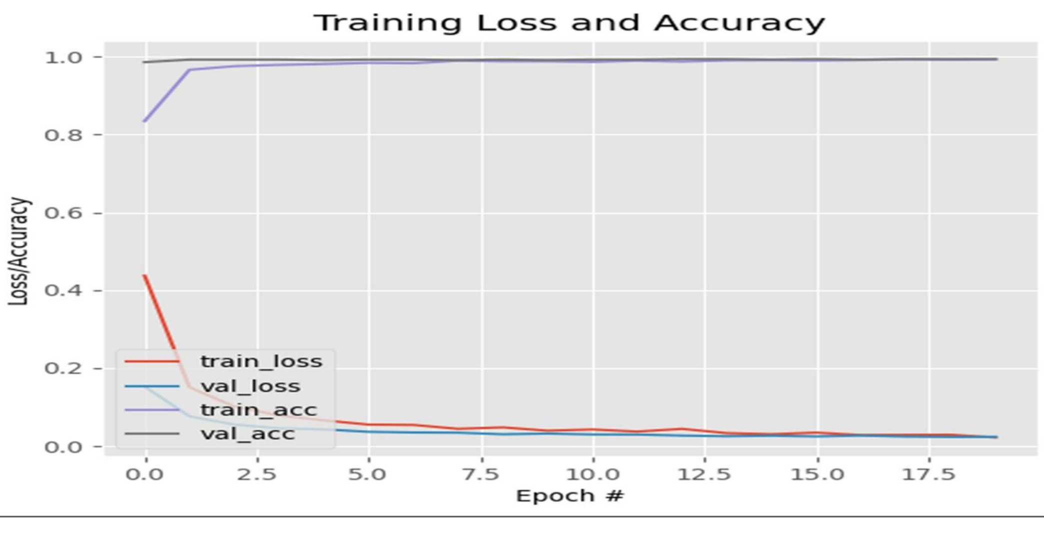
The fit method trains the model using backpropagation to minimize the loss between the predicted output of the model and the actual output and evaluates its performance on the validation data after each epoch.

The predict method returns an array of predicted probabilities for each input in the testing dataset.

Accuracy is set as the evaluation metric to be used during training and testing.

Binary cross-entropy loss function calculates the difference between the predicted probabilities and the true labels, and minimizes this difference during training.

Training loss and Accuracy:-



Detection and Prediction:-

The function takes an input frame, a face detection network and a face mask detection network and returns a list of face locations and their corresponding predictions of wearing a mask.

Bounding boxes and Labels:-

For each frame, we call our function to detect the face locations and mask predictions and then draws bounding boxes and labels on the frame based on the predictions.

Finally, it displays the output frame in a window.

Social Distancing:-

We calculate the Euclidean distance between 2 persons(objects) and also take into consideration the distance from the camera of the 2 objects, if it is less than a particular distance, then there is a violation of social distancing and thus, we give a warning by giving red bounding box around it. If it greater, then green bounding box is displayed.

References-

https://ieeexplore.ieee.org/document/9325631

Under the Supervision of

Dr. Aruna Tiwari

Professor, CSE