Project:Face Mask Detection

Abstract:COVID-19 pandemic has rapidly affected our day-to-day life disrupting world trade and movements. Wearing a protective face mask has become a new normal. In the near future, many public service providers will ask the customers to wear masks correctly to avail of their services. Therefore, face mask detection has become a crucial task to help global society.This project helps us to spread awareness among people using face masks properly. It detects the face mask on your face whether the person is hiding his/her face by mask or not.

Programing Language Used:Python

Technologies Used: OpenCV, CNN

Steps Followed:1)Importing Libraries

2)Loading Data

3)Data Preprocessing

4)Splitting Dataset

5)Training Model

6)Prediction on unknown data image,live detection through webcam

7)Evaluation

About The Model MobileNetv2:

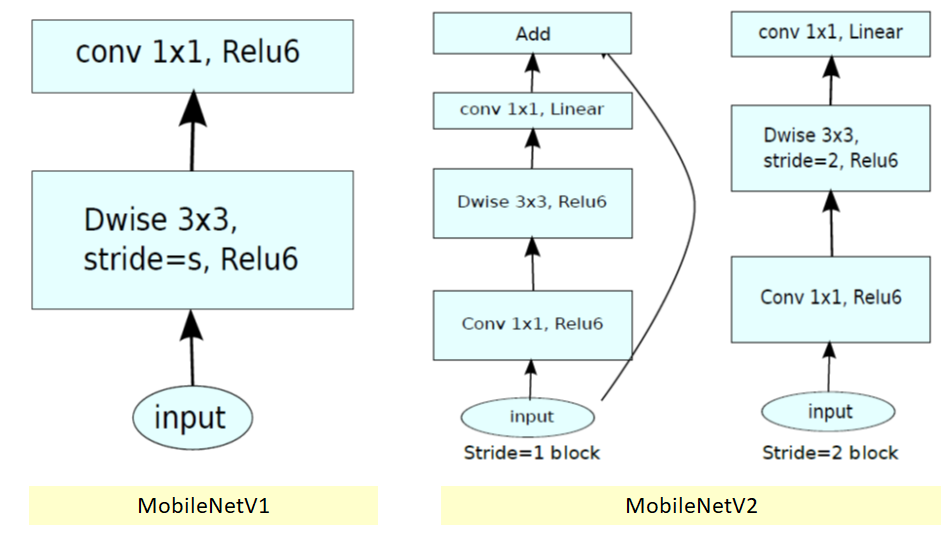
MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. It is based on an inverted residual structure where the residual connections are between the bottleneck layers.

This function returns a Keras image classification model, optionally loaded with weights pre-trained on ImageNet.

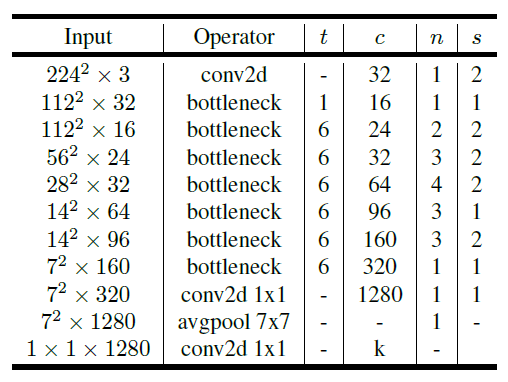
The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers.

* In MobileNetV2, there are two types of blocks. One is a residual block with stride of 1. Another one is block with stride of 2 for downsizing.
* There are 3 layers for both types of blocks.
* The first layer is 1×1 convolution with ReLU6**.**
* The second layer is the depthwise convolution.
* The third layer is another 1×1 convolution but without any non-linearity. It is claimed that if ReLU is used again, the deep networks only have the power of a linear classifier on the non-zero volume part of the output domain.
* If the input got 64 channels, the internal output would get 64×*t*=64×6=384 channels.

MobileNetV2 Convolutional Blocks:



# Overall Architecture



MobileNetV2 Overall Architecture

* where *t*: expansion factor
* *c*: number of output channels
* *n*: repeating number
* s: stride. 3×3 kernels are used for spatial convolution.
* In typical , the primary network (width multiplier 1, 224×224), has a computational cost of 300 million multiply-adds and uses 3.4 million parameters.

Here we can see the MobileNetV2 building block:

