IJARSCT
ISSN (Online) 2581-9429
Impact Factor: 6.252
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 1, June 2022

Face Mask Detection

Sudhakar P. Rupnar[1], Prathamesh Gujjalwar[2], Abhishek Aradhye[3], Chandroday Kalshetty[4]
Shubham Gade[5], Mr. V. M. Sale[6]
Department of Computer Science and Engineering
SVERI's College of Engineering, Pandharpur, India

Abstract: *The corona virus COVID-19 pandemic is causing a global health crisis so the effective protection methods is wearing a face mask in public areas according to the World Health Organization (WHO). The COVID-19 pandemic forcedgovernments across the world to impose lockdowns to prevent virus transmissions. Reports indicate that wearing face maskswhile at work clearly reduces the risk of transmission. An efficient and economic approach of using AI to create a safe environment in a manufacturing setup. A hybrid model usingdeep and classical machine learning for face mask detection will be presented. A face mask detection dataset consists of with mask and without mask images , we are going to use OpenCV to do real-time face detection from a live stream viaour webcam. We will use the dataset to build a COVID-19 facemask detector with computer vision using Python, OpenCV, and Tensor Flow and Keras. Our goal is to identify whether theperson on image/video stream is wearing a face mask or notwith the help of computer vision and deep learning.*

Keywords: Deep Learning, Computer Vision, OpenCV, Tensorflow, Keras.

I. INTRODUCTION

The trend of wearing face masks in public is rising due to theCOVID- 19 corona virus epidemic all over the world. Before Covid-19, People used to wear masks to protect their healthfrom air pollution. While other people are self-conscious about their looks, they hide their emotions from the public by hiding their faces. Scientists proofed that wearing face masks works on impeding COVID-19 transmission. COVID-19 (known as corona virus) is the latest epidemic virus thathit the human health in the last century. In 2020, the rapid spreading of COVID-19 has forced the World HealthOrganization to declare COVID- 19 as a global pandemic.

More than five million cases were infected by COVID-19 in less than 6 months across 188 countries. The virus spreads through close contact and in crowded and overcrowded areas.

The corona virus epidemic has given rise to an extraordinary degree of worldwide scientific cooperation. Artificial Intelligence (AI) based on Machine learning and Deep Learning can help to fight Covid-19 in many ways. Machine learning allows researchers and clinicians evaluate vastquantities of data to forecast the distribution of COVID-19, to serve as an early warning mechanism for potential pandemics, and to classify vulnerable populations.The provision of healthcare needs funding for emerging technology such as artificial intelligence, IoT, big data and machine learning to tackle and predict new diseases. In order to better understand infection rates and to trace and quickly detect infections, the AI's power is being exploited toaddress the Covid-19 pandemic. People are forced by laws to wear face masks in public in many countries. These rules and laws were developed as an action to the exponential growthin cases and deaths in many areas. However, the process ofmonitoring large groups of people is becoming more difficult. The monitoring process involves the detection of anyone who is not wearing a face mask.

Here we introduce a mask face detection model that is based on computer vision and deep learning. The proposed model can be integrated with surveillance cameras to impede the COVID-19 transmission by allowing the detection of people who are wearing masks not wearing face masks. The modelis integration between deep learning and classical machine learning techniques with opencv, tensor flow and keras. Wehave used deep transfer leering for feature extractions and combined it with three classical machine learning algorithms. We introduced a comparison between them to find the most suitable algorithm that achieved the highest accuracy and consumed the least time in the process of training and detection.

1.1 Machine Learning

Machine learning (**ML**) is the study of computer algorithms that improve automatically through experience. It is seen asa subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictionsor decisions without being explicitly programmed to do so.

Machine learning algorithms are used in a wide variety of applications, such as email filtering and computer vision, where it is difficult or infeasible to develop conventional algorithms to perform the needed tasks. Machine learning is closely related to computational statistics, which focuses onmaking predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning. In its application across business problems, machine learning is also referred to as predictive analytics.

Machine learning approaches are traditionally divided intothree broad categories, depending on the nature of the"signal" or "feedback" available to the learning system:

- Supervised learning: The computer is presented with example inputs and their desired outputs, given bya "teacher", and the goal is to learn a general rule that maps inputs to outputs.

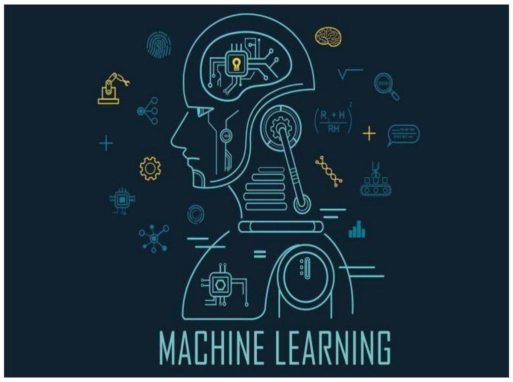


Fig -1: **Machine learning outlook**

- Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or ameans towards an end (feature learning).
- Reinforcement learning: A computer program interacts with a dynamic environment in which it mustperform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates itsproblem space, the program is provided feedback that'sanalogous to rewards, which it tries to maximize.

Other approaches have been developed which don't fit neatly into this three-fold categorization, and sometimes more than one is used by the same machine learning system.

1.2 Computer Vision

Computer vision is an interdisciplinary scientific field thatdeals with how computers can gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to understand and automate tasks thatthe human visual system can do, Computer vision tasks include methods for acquiring, processing, analyzing and understanding digital images, and extraction of high- dimensional data from the real world in order to produce numerical or symbolic information, e.g. in the forms of decisions, Understanding in this context means the transformation of visual images (the input of the retina) intodescriptions of the world that make sense to thought processes and can elicit appropriate action. This image understanding can be seen as the disentangling of symbolicinformation from image data using models constructed withthe aid of geometry, physics, statistics, and learning theory.

The scientific discipline of computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, multi-dimensional data from a 3D scanner or medical scanning device. The technological discipline of computer vision seeks to apply its theories and models to the construction of computer vision systems. Computer vision is an interdisciplinary field that deals with how computers can be made to gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to automate tasks that the human visual system can do.

Fig – 2: Computer Vision

Computer vision is concerned with the automatic extraction,analysis and understanding of useful information from a single image or a sequence of images. It involves the development of a theoretical and algorithmic basis to achieve automatic visual understanding. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images.
The image data can take many forms, such as video sequences, views from multiple cameras, or multi- dimensional data from a medical scanner. As a technologicaldiscipline, computer vision seeks to apply its theories and models for the construction of computer vision systems.

1.3 Deep Learning

Deep learning methods aim at learning feature hierarchieswith features from higher levels of the hierarchy formed bythe composition of lower level features. Automatically learning features at multiple levels of abstraction allow a system to learn complex functions mapping the input to theoutput directly from data, without depending completely onhuman-crafted features. Deep learning algorithms seek to exploit the unknown structure in the input distribution in order to discover good representations, often at multiple levels, with higher-level learned features defined in terms of lower-level features.

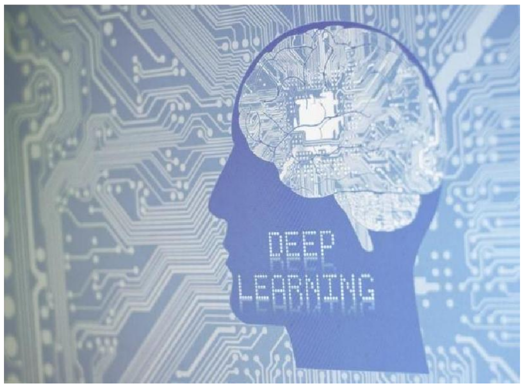


Fig 3: Deep learning

The hierarchy of concepts allows the computer to learn complicated concepts by building them out of simpler ones.If we draw a graph showing how these concepts are built on top of each other, the graph is deep, with many layers. For this reason, we call this approach to AI deep learning. Deeplearning excels on problem domains where the inputs (andeven output) are analog. Meaning, they are not a few quantities in a tabular format but instead are **images of pixel data, documents of text data or files of audio data.**Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction.

1.4 OpenCV

OpenCV (Open Source Computer Vision Library) is an opensource computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercialproducts. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, whichincludes a comprehensive set of both classic and state-of- the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces,identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image ofan entire scene, find similar images from an image database,remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. OpenCV has more than 47 thousand people of user community and estimatednumber of downloads exceeding 18 million. The library is used extensively in companies, research groups and by governmental bodies.Along with well-established companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota that employ the library, there are many startups such as Applied Minds, Video Surf, and Zeitera, that make extensiveuse of OpenCV. OpenCV's deployed uses span the range fromstitching street view images together, detecting intrusions in surveillance video in Israel, monitoring mine equipment in China, helping robots navigate and pick up objects at WillowGarage, detection of swimming pool drowning accidents in Europe, running interactive art in Spain and New York, checking runways for debris in Turkey, inspecting labels onproducts in factories around the world on to rapid face detection in Japan.

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured CUDA and OpenCL interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or supportthose algorithms. OpenCV is written natively in C++ and hasa template interface that works seamlessly with STL containers.

1.5 Tensor Flow

TensorFlow is a free and open-source software library fordataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google, TensorFlow is Google Brain's second-generation system. Version 1.0.0 was released on February 11, While the reference implementation runs on single devices, TensorFlow can run on multiple CPUs and GPUs (with optional CUDA and SYCL extensions for general-purpose computing on graphics processing units).

Tensor Flow is available on 64-bit Linux, macOS, Windows,and mobile computing platforms including Android and iOS. Its flexible architecture allows for the easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices.

The name Tensor Flow derives from the operations that such neural networks perform on multidimensional data arrays, which are referred to as *tensors*. During the Google I/O Conference in June 2016, Jeff Dean stated that 1,500 repositories on GitHub mentioned TensorFlow, of which only 5 were from Google.Unlike other numerical libraries intended for use in Deep Learning like Theano, TensorFlowwas designed for use both in research and development andin production systems, not least RankBrain in Google searchand the fun DeepDream project.It can run on single CPU systems, GPUs as well as mobile devices and large scale distributed systems of hundreds of machines.

1.6 KERAS

Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number ofuser actions required for common use cases, and it provides clear & actionable error messages. It also has extensive documentation and developer guides. Keras contains numerous implementations of commonly used neural- network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. The code is hosted on GitHub, and community support forums include the GitHub issues page, and a Slack channel. Keras is a minimalist Python library for deep learning that can run ontop of Theano or Tensor Flow. It was developed to make implementing deep learning models as fast and easy as possible for research and development. It runs on Python 2.7 or 3.5 and can seamlessly execute on GPUs and CPUs giventhe underlying frameworks. It is released under thepermissive MIT license.

Keras was developed and maintained by François Chollet, aGoogle engineer using four guiding principles:

- **Modularity**: A model can be understood as a sequence or a graph alone. All the concerns of a deep learning model are discrete components thatcan be combined in arbitrary ways.
- **Minimalism**: The library provides just enough to achieve an outcome, no frills and maximizing readability.
- **Extensibility**: New components are intentionally easy to add and use within the framework, intended for researchers to trial and explore new ideas.
- **Python**: No separate model files with custom file formats. Everything is native Python. Keras is designed for minimalism and modularity allowingyou to very quickly define deep learning models and run them on top of a Theano or TensorFlow backend.

1.7 PyTorch

PyTorch is an open source machine learning library basedon the Torch library, used for
applications such as computervision and natural language processing, primarily developedby Face book's AI Research lab (FAIR).

It is free and open-source software released under the Modified BSD license. Although the Python interface is more polished and the primary focus of development, PyTorch also has a C++ interface. Tensor computing (like NumPy) with strong acceleration via graphics processing units (GPU).

Deep neural networks built on a tape-based automatic differentiation system

PyTorch defines a class called Tensor (torch.Tensor) to store and operate on homogeneous multidimensional rectangular arrays of numbers. PyTorch Tensors are similar to NumPy Arrays, but can also be operated on a CUDA-capable Nvidia GPU. PyTorch supports various sub-types of Tensors.

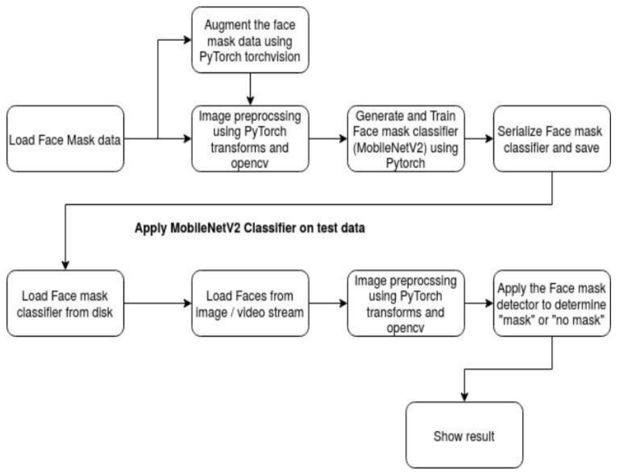
II. PROPOSED SYSTEM

The proposed system focuses on how to identify the person on image/video stream wearing face mask withthe help of computer vision and deep learning algorithm by using the OpenCV, Tensor flow, Keras and PyTorch library.

Approach

1. Train Deep learning model (MobileNetV2)
2. Apply mask detector over images / live video stream

Flow Chart


Augment the face mask data using PyTorch torchvision
Load Face Mask data
Image preprocssing using PyTorch transforms and opencv
Generate and Train Face mask classifier (MobileNetV2) using Pytorch
Serialize Face mask classifier and save
Apply MobileNetV2 Classifier on test data
Load Face mask classifier from disk
Load Faces from image / video stream
Image preprocssing using PyTorch transforms and opencv
Apply the Face mask detector to determine "mask" or "no mask"
Show result

Data at Source

The majority of the images were augmented by OpenCV. The set of images were already labeled "mask" and "no mask". The images that were present were of different sizes and resolutions, probably extracted from different sources or from machines (cameras) of different resolutions.

Data preprocessing

preprocessing steps as mentioned below was applied to allthe raw input images to convert them into clean versions, which could be fed to a neural network machine learning model.

1. Resizing the input image (256 x 256)
2. Applying the color filtering (RGB) over the channels (Ourmodel MobileNetV2 supports 2D 3 channel image)
3. Scaling / Normalizing images using the standard mean ofPyTorch buildin weights
4. Center cropping the image with the pixel value of 224x224x3
5. Finally Converting them into tensors (Similar to NumPy array)

Deep Learning Frameworks

To implement this deep learning network we have the following options.

1. Tensor Flow
2. Keras
3. PyTorch
4. Caffee
5. MxNet
6. Microsoft Cognitive Tool Kit

We are using the PyTorch because it runs on Python, which means that anyone with a basic understanding of Python can get started on building their deep learning models, and also it has the following advantage compared with Tensor Flow

1. Data Parallelism
2. It looks like a Framework.

MobileNetV2

MobileNetV2 builds upon the ideas from MobileNetV1, using depth wise separable convolution as efficient building blocks. However, V2 introduces two new features to the architecture:

1. Linear bottlenecks between the layers, and
2. Shortcut connections between the bottlenecks.

The basic structure is shown below

The typical MobilenetV2 architecture has as many layers listed below, In Pytorch we can use the models library in TorchVision to create the MobileNetV2 model instead of defining/building our own model. The weights of each layer in the model are predefined based on the ImageNet dataset. The weights indicate the padding, strides, kernel size, input channels and output channels.MobileNetV2 was chosen as an algorithm to build a model that could be deployed on a mobile device. A customized fully connected layer which contains four sequential layers on top of the MobileNetV2 model was developed.

The layers are

1. Average Pooling layer with 7×7 weights
2. Linear layer with ReLu activation function
3. Dropout Layer
4. Linear layer with Softmax activation function with the result of 2 values. The final layer softmax function gives the result of two probabilities each one represents the classification of"mask" or "not mask".

Face Mask Detection in webcam stream The flow to identify the person in the webcam wearing the face mask or not. The process is two-fold.

1. **To identify the faces in the webcam**
2. **Classify the faces based on the mask Identify the Face in the Webcam**

To identify the faces a pre-trained model provided by the OpenCV framework was used. The model was trained using web images. OpenCV provides 2 models for this face detector:

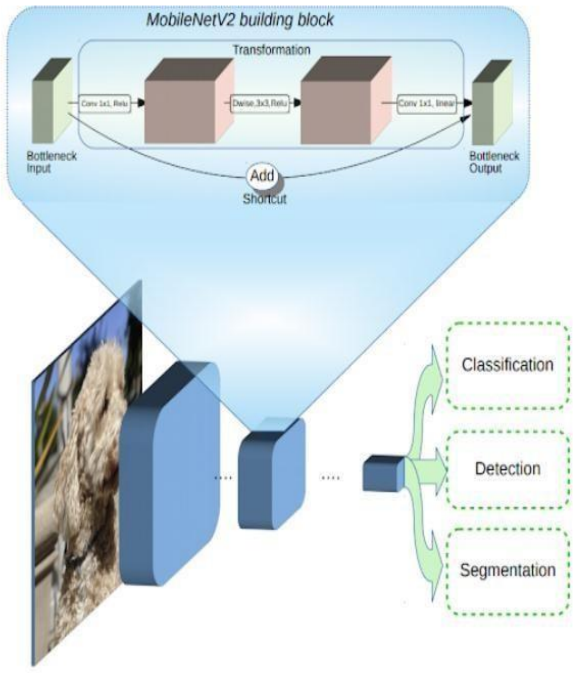


Fig 4: Detection of "Mask" and "No Mask"

1. Floating-point 16 version of the original Caffe implementation.
2.8 bit quantized version using Tensor flow

The Caffe model in this face mask detector. There has been a lot of discussion around deep learning based approaches for person detection. This encouraged us to come up with our own algorithm to solve this problem.

Our work on face mask detection comprises of data collection to tackle the variance in kinds of face masks worn by the workers. The face mask detection model is a combination of face detection model to identify the existing faces from camera feeds and then running those faces through a mask detection model.

III. CONCLUSION

As the technology are blooming with emerging trends the availability so we have novel face mask detector which can possibly contribute to public healthcare. The architecture consists of Mobile Net as the backbone it can be used for high and low computation scenarios. In order to extract more robust features, we utilize transfer learning to adopt weights from a similar task face detection, which is trained on a very large dataset.

We used OpenCV, tensor flow, keras , Pytorch and CNN to detect whether people were wearing face masks or not. The models were tested with images and real-time video streams. The accuracy of the model is achieved and, the optimization of the model is a continuous process and we are building a highly accurate solution by tuning the hyper parameters. This specific model could be used as a use case for edge analytics.

Furthermore, the proposed method achieves state-of-the-art results on a public face mask face masks or not. Themodels were tested with images and real-time video streams. The accuracy of the model is achieved and, the optimization of the model is a continuous process and we are building a highly accurate solution by tuning the hyper parameters.

This specific model could be used as a use casefor edge analytics. Furthermore, the proposed method achieves state-of-the-art results on a public face mask dataset. By the development of face mask detection we candetect if the person is wearing a face mask and allow their entry would be of great help to the society.

REFERENCES

[1]. P. A. Rota, M. S. Oberste, S. S. Monroe, W. A. Nix, R. Campagnoli, J. P. Icenogle, S. Penaranda, B. Bankamp,K. Maher, M.-h. Chenet al., "Characterization of a novel coronavirus associated with severe acute respiratorysyndrome,"science, vol. 300, no. 5624, pp. 1394–1399, 2003.

[2]. Z. A. Memish, A. I. Zumla, R. F. Al-Hakeem, A. A. Al- Rabeeah, and G. M. Stephens, "Family cluster of middleeast respiratory syndrome coronavirus infections,"New England Journal of Medicine, vol. 368, no. 26, pp.2487–2494, 2013.

[3]. Y. Liu, A. A. Gayle, A. Wilder-Smith, and J. Rocklöv, "The reproductive number of covid-19 is higher comparedto sars coronavirus,"Journal of travel medicine, 2020.

[4]. Y. Fang, Y. Nie, and M. Penny, "Transmission dynamics of the covid-19 outbreak and effectiveness of governmentinterventions: A data- driven analysis,"Journal of medical virology, vol. 92, no. 6, pp. 645–659, 2020.

[5]. N. H. Leung, D. K. Chu, E. Y. Shiu, K.-H. Chan, J. J. McDevitt, B. J. Hau, H.-L. Yen, Y. Li, D. KM, J. Ipet al.,"Respiratory virus shedding in exhaled breath and efficacy of face masks."

[6]. S. Feng, C. Shen, N. Xia, W. Song, M. Fan, and B. J. Cowling, "Rational use of face masks in the covid-19pandemic,"The Lancet Respiratory Medicine, 2020.

[7]. Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Peiet al., "Masked facerecognition dataset and application,"arXiv preprint arXiv:2003.09093, 2020.[10]Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review,"IEEE transactions on neural networks and learning systems, vol. 30, no. 11, pp. 3212–3232, 2019.

[8]. A. Kumar, A. Kaur, and M. Kumar, "Face detection techniques: a review,"Artificial Intelligence Review, vol. 52,no. 2, pp. 927–948, 2019.D.-H. Lee, K.-L. Chen, K.-H. Liou, C.-L. Liu, and J.-L. Liu, "Deep learning and control algorithms of direct perception for autonomous driving,2019.