FACE MASK DETECTION SYSTEM USING COMPUTER VISION AND DEEP LEARNING

This project report is submitted to

Silicon Institute of Technology, Bhubaneswar

in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

Submitted by

Nilima Satapathy (1701209221)

Dhanalaxmi Nayak (1701209112)

Group No. : CPRC-4

Under the Esteemed Supervision of

Prof. Dr. Soumya Ranjan Samal

Prof. Mr. Tarini Charana Mishra

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SILICON INSTITUTE OF TECHNOLOGY

SILICON HILLS, BHUBANESWAR – 751024, ODISHA, INDIA

April, 2021

CERTIFICATE

This is to certify that the work contained in the project entitled **"Face mask detection using computer Vision and deep learning"**, submitted by **Nilima Satapathy, (Regd. No.: 1701209221), Dhanalaxmi Nayak (Regd. No.: 1701209112)** is a record of bonafide works carried out by them under my supervision and guidance. The contents embodied in the project is being submitted as a part of 8th semester project for the undergraduate curriculum and have not been submitted for the award of any other degree or diploma in this or any other university.

Date : 31.03.2021

Place: Bhubaneswar

Prof. Tarini Charan Mishra
Professor
Department of Computer Science & Engineering

External Examiner

Silicon
...beyond teaching

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SILICON INSTITUTE OF TECHNOLOGY
BHUBANESWAR – 751024

DECLARATION

We hereby certify that:-

a. The work contained in the project is original and has been done by ourselves under the supervision of our supervisor.

b. The work has not been submitted to any other Institute for any degree or diploma.

c. We have conformed to the norms and guidelines given to us by the Project Review Committee of our department.

d. Whenever we have used materials (data, theoretical analysis and text) from other sources, we have given due credit to them by citing them in the text of the project and giving their details in the references.

Date : 31.03.2021

Place: Bhubaneswar

Nilima Satapathy (1701209221) ……...

Dhanalaxmi Nayak (1701209112) ……...

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SILICON INSTITUTE OF TECHNOLOGY
BHUBANESWAR – 751024

ACKNOWLEDGEMENTS

We take upon this opportunity to acknowledge the many people whose guidance and mentorship has helped us throughout. We am deeply indebted to my supervisor and mentor **Prof. Tarini Charan Mishra**. We would like to thank to **Dr Soumya Ranjan Samal** for his guidance and constant encourangement. We further thank to our HOD**, Dr Pradyumna Tripat**hy and **Prof. Bikram Kesari Mishra**(Project Coordinator)for their immense co-operation and support.

We owe our sincere gratitude towards our family for providing us with all the financial assistance. Our heartfelt thanks to our friends for helping us throughout and also motivating us. We also express my deepest gratitude to all those who have directly and indirectly contributed towards the successful completion of this project.

Nilima Satapathy.....

Dhanalaxmi Nayak.....

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SILICON INSTITUTE OF TECHNOLOGY
BHUBANESWAR – 751024

Abstract

The corona virus COVID-19 pandemic is causing a global health crisis so the effective protection method is wearing a face mask in public areas according to the World Health Organization (WHO). The COVID-19 pandemic forced governments across the world to impose lockdowns to prevent virus transmissions. Reports indicate that wearing face masks while 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 using deep 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 via our webcam. We will use the dataset to build a COVID-19 face mask detector with computer vision using Python, OpenCV, and Tensor Flow and Keras. Our goal is to identify whether the person on image/video stream is wearing a face mask or not with the help of computer vision and deep learning

Keywords: Open CV, Tensor Flow, Keras, Deep Learning ,Machine Learning, Real time Face Mask detection.

LIST OF ABBREVIATIONS

Abbreviation	Description
ASM	Accelerated Simulation Mode
CNN	Convolutional Neural Network
DL	Deep Learning
GPU	Graphics Processing Unit
IR	Infra Red
LDA	Linear Discriminate Analysis
ML	Machine Learning
PDM	Point Distribution Model
PCA	Principal Component Analysis
SSD	Single Shot Detector
YOLO	You Only look Once

LIST OF FIGURES

CONTENTS

CHAPTER 1

INTRODUCTION

The trend of wearing face masks in public is rising due to the COVID- 19 corona virus epidemic all over the world. Before Covid-19, People used to wear masks to protect their health from 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 that hit the human health in the last century. In 2020, the rapid spreading of COVID-19 has forced the World Health Organization to declare COVID- 19 as a global pandemic.

A novel coronavirus has resulted in person-to-person transmission but as far as we know, the transmission of the novel coronavirus causing coronavirus disease 2019 (COVID-19) can also be from an asymptomatic carrier with no covid symptoms. It has spread rapidly across the world, bringing massive health, economic, environmental and social challenges to the entire human population. At the moment, WHO recommends that people should wear face masks to avoid the risk of virus transmission and also recommends that a social distance of at least 2m be maintained between individuals to prevent person-to- person spread of disease. Furthermore, many public service providers require customers to use the service only if they wear masks and follow safe social distancing.

Therefore, face mask detection and safe social distance monitoring has become a crucial computer vision task to help the global society. This paper describes approach to prevent the spread of the virus by monitoring in real time if person is following safe social distancing and wearing face masks in public places.

This paper adopts a combination of Deep Learning and Computer Vision with transfer learning technique to achieve the balance of resource limitations and recognition accuracy so that it can be used on real-time video surveillance to monitor public places to detect if persons wearing face mask and maintaining safe social distancing.

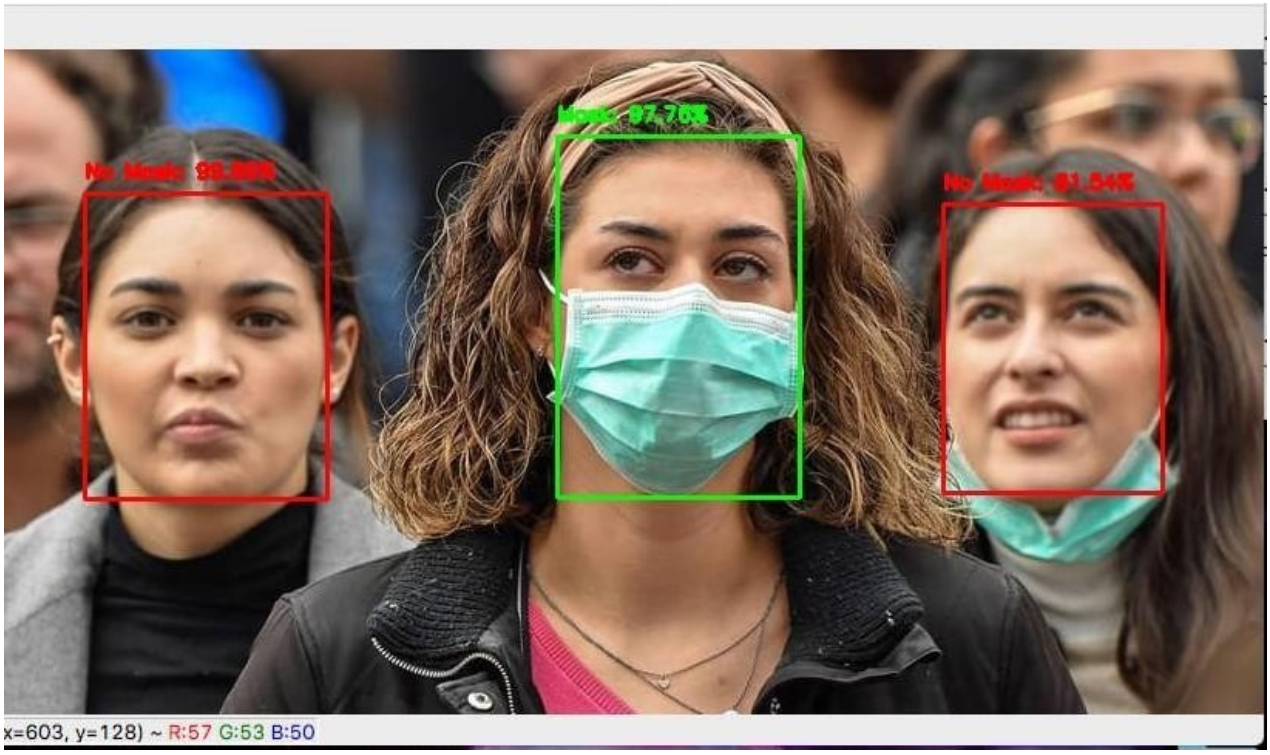

Fig- 1.1: Basic structure of Face Mask Detection Model

Our solution uses neural networking models to analyze Real-Time Streaming Protocol (RTSP) video streams using OpenCV and MobileNet V2. We mix the approach of modern-day deep learning and classic projective geometry techniques which not only helps to meet the real-time requirements, but also keeps high prediction accuracy. If the person detected as not following the covid-19 safety guidelines, violation alerts will be send to the control center at state police headquarters for taking further action. It allows automating the solution and enforces the wearing of the mask and follows the guidelines of social distancing. This model was created to run on python and the accuracy obtained was between 85% and 95%.

MACHINE LEARNING:

Machine learning (ML)is the study of computer algorithms thatimprove automatically through experience. Itis seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or 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 on making 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 into three 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 by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
- 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 a means towards an end (feature learning).
- Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous 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

COMPUTER VISION :

Computer vision is an interdisciplinary scientific field that deals 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 that the human visual system can do, Computer vision tasks include methods for acquiring, processing, analyzing and understanding digital images, and extraction of highdimensional 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 ofthe retina)into descriptions of the world that make sense to thought processes and can elicit appropriate action. This image understanding can be seen as the disentangling of symbolic information from image data using models constructedwith the 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, multidimensional 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.

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 multidimensional data from a medical scanner. As a technological discipline, computer vision seeks to apply its theories and models for the construction of computer vision systems.

DEEP LEARNING:

Deep learning methods aim at learning feature hierarchies with features from higher levels of the hierarchy formed by the composition of lower level features. Automatically learning features at multiple levels of abstraction allow a system to learn complex functions mapping the input to the output directly from data, without depending completely on human-crafted features. Deep learning algorithms seek to exploit the unknown structure in the input distribution in order to discover good representations, often at multiple.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. Deep learning excels on problem domains where the inputs (and even 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.

OpenCV:

OpenCV (Open Source Computer Vision Library) is an open source 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 commercial products. 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,which includes 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 of an 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 estimated number 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 extensive use of OpenCV. OpenCV's deployed uses span the range from stitching street view images together, detecting intrusions in surveillance video in Israel, monitoring mine equipment in China, helping robots navigate and pick up objects at Willow Garage, detection of swimming pool drowning accidents in Europe, running interactive art in Spain and New York, checking runways for debris in Turkey, inspecting labels on products 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 support those algorithms. OpenCV is written natively in C++ and has a template interface that works seamlessly with STL containers.

1.1. BACKGROUND

At the moment, WHO recommends that people should wear face masks to avoid the risk of virus transmission and also recommends that a social distance of at least 2m be maintained between individuals to prevent person-to- person spread of disease. Furthermore, many public service providers require customers to use the service only if they wear masks and follow safe social distancing. Therefore, face mask detection and safe social distance monitoring has become a crucial computer vision task to help the global society. This paper describes approach to prevent the spread of the virus by monitoring in real time if person is following safe social distancing and wearing face masks in public places.

1.2. PROBLEM STATEMENT

This pandemic has left us no other choice but to wear a mask and maintain social distancing along with sanitation measures until the vaccine is fully developed and distributed. However implementation & regulation of such sanitation measures and personal protection gears like masks, sanitisers demands a lots of management related issues. It's very difficult to monitor people wearing mask or people without mask in large scale , It requires a lot of human resources to constantly monitor a lot of people.

1.3. OBJECTIVE AND MOTIVATION

Face mask detection and safe social distance monitoring has become a crucial computer vision task to help the global society. This paper describes approach to prevent the spread of the virus by monitoring in real time if person is following safe social distancing and wearing face masks in public places.

1.4. PROPOSED METHOD

This paper adopts a combination of Deep Learning and Computer Vision with transfer learning technique to achieve the balance of resource limitations and recognition accuracy so that it can be used on real-time video surveillance to monitor public places to detect if

persons wearing face mask and maintaining safe social distancing.

A Convolutional Neural Network is an algorithm that can take in an image as input, assign importance (in the form of trainable weights and biases) to aspects or features of the image and output a decision or some other form of logic based on what it has "seen" in the image. The data pre- processing / preparation required is very minimal when it comes to CNN architectures as the features of your data are the actual pixel values and with enough training data, CNN will automatically define which of these pixels are most important in itsdecision-making.

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.MobileNetV2was 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”.

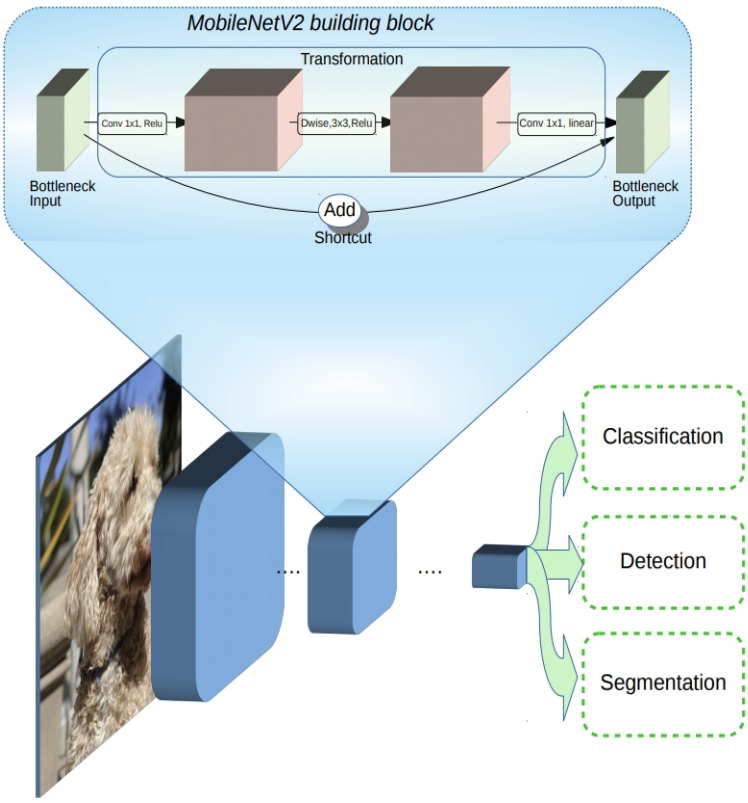

Figure: 1.2. MobileNetV2 model

CHAPTER 2

LITERATURE REVIEW

This section deals with various literature work done under the topic of Face mask detection model (using deep learning & various methods)

2.1. DIFFERENT APPROACHES OF FACE RECOGNITION:

There are two predominant approaches to the face recognition problem: Geometric (feature based) and photometric (view based). As researcher interest in face recognition continued, many different algorithms were developed, three of which have been well studied in face recognition literature. Recognition algorithms can be divided into two main approaches: 1. Geometric: Is based on geometrical relationship between facial landmarks, or in other words the spatial configuration of facial features.

That means that the main geometrical features of the face such as the eyes, nose and mouth are first located and then faces are classified on the basis of various geometrical distances and angles between features. (Figure 3) 2. Photometric stereo: Used to recover the shape of an object from a number of images taken under different lighting conditions. The shape of the recovered object is defined by a gradient map, which is made up of an array of surface normals (Zhao and Chellappa, 2006) (Figure 2) Popular recognition algorithms include:

Principal Component Analysis using Eigenfaces, (PCA)

Linear Discriminate Analysis,

Elastic Bunch Graph Matching using the Fisherface algorithm,

In recent years, object detection techniques using deep models are potentially more capable than shallow models in handling complex tasks and they have achieved spectacular progress in computer vision. Deep models for person detection focus on feature learning contextual information learning, and occlusion handling. Deep learning object detection models can now mainly be divided into two families:

two-stage detectors such as R-CNN, Fast R-CNN and Faster R-CNN and their variants and one-stage detectors such as YOLO and SSD. In two-stage detectors detection is performed in

stages, in the first stage, computed proposals and classified in the second stage into object

categories. However, some methods, such as YOLO, SSD MultiBox, consider detection as a regression issue and look at the image once for detection.

In proposed system we are using Single Shot Detector MultiBox(SSD) which seems to be a good choice for real-time object detection and the accuracy trade-off is also very little. SSD uses the VGG-16 model pre-trained on ImageNet as its basic model to extract useful image feature. At the top of VGG16, SSD adds several convolutional feature layers of decreasing sizes.

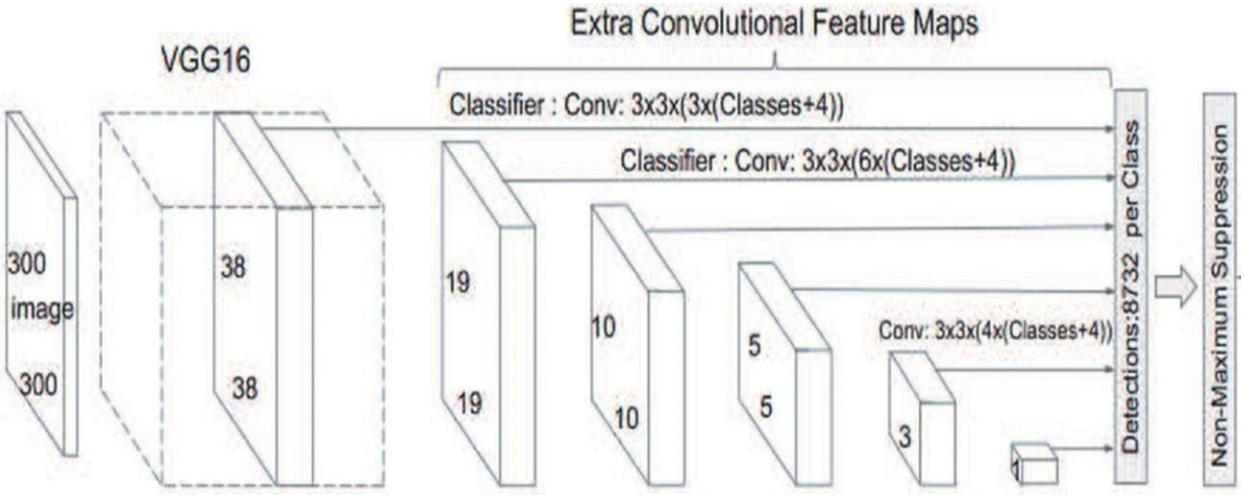

Figure 2.1: Model Architecture of SSD

The Viola – Jones object detection system can be trained to detect any object, but is especially common for facial detection and is more accurate and faster. The Viola and Jones process is an example of supervised learning. Zhu also shared another very widespread facial detection algorithm is a neural network-based detector.

It only works well with the front, upright face. Li et al. suggested another model for facial detection which was a Multi- View Face Detector with surf capabilities. Oro et al. also proposed a haar-like feature based face detection algorithm for HD video on the GTX470 and obtained an improved speed of 2.5 times. However, they only used CUDA which is a GPU programming tool for NVIDIA GPUs.

Compared to OpenCL which is used in a number of computed components, it is unable to resolve the imbalanced workload issue experienced during the implementation of the viola-jones face detection algorithm in GPUs. Glass et al. addressed the importance of social differencing and how the risk of pandemic growth can be slowly decreased by successfully preserving social distance without the use of vaccines or antiviral drugs.

The authors have carried out an exhaustive study on this in both rural and urban communities in order to demonstrate a reduction in the growth rate. Z., Luo studies the identification of people with full-face or partial occlusion. This approach categorizes into way, people with hand over their faces or occluded with objects.

This approach is not suited to our scenario, which requires, in essentially, to detect faces that have their mouths covered with masks such as scarves, mufflers, handkerchiefs, etc

CHAPTER 3

METHODOLOGY

3.1. INTRODUCTION

The proposed system helps to ensure the safety of the people at public places by automatically monitoring them whether they maintain a safe social distance, and also by detecting whether or not and individual wears face mask. This section briefly describes the solution architecture and how the proposed system will automatically functions in an automatic manner to prevent the corona virus spread.

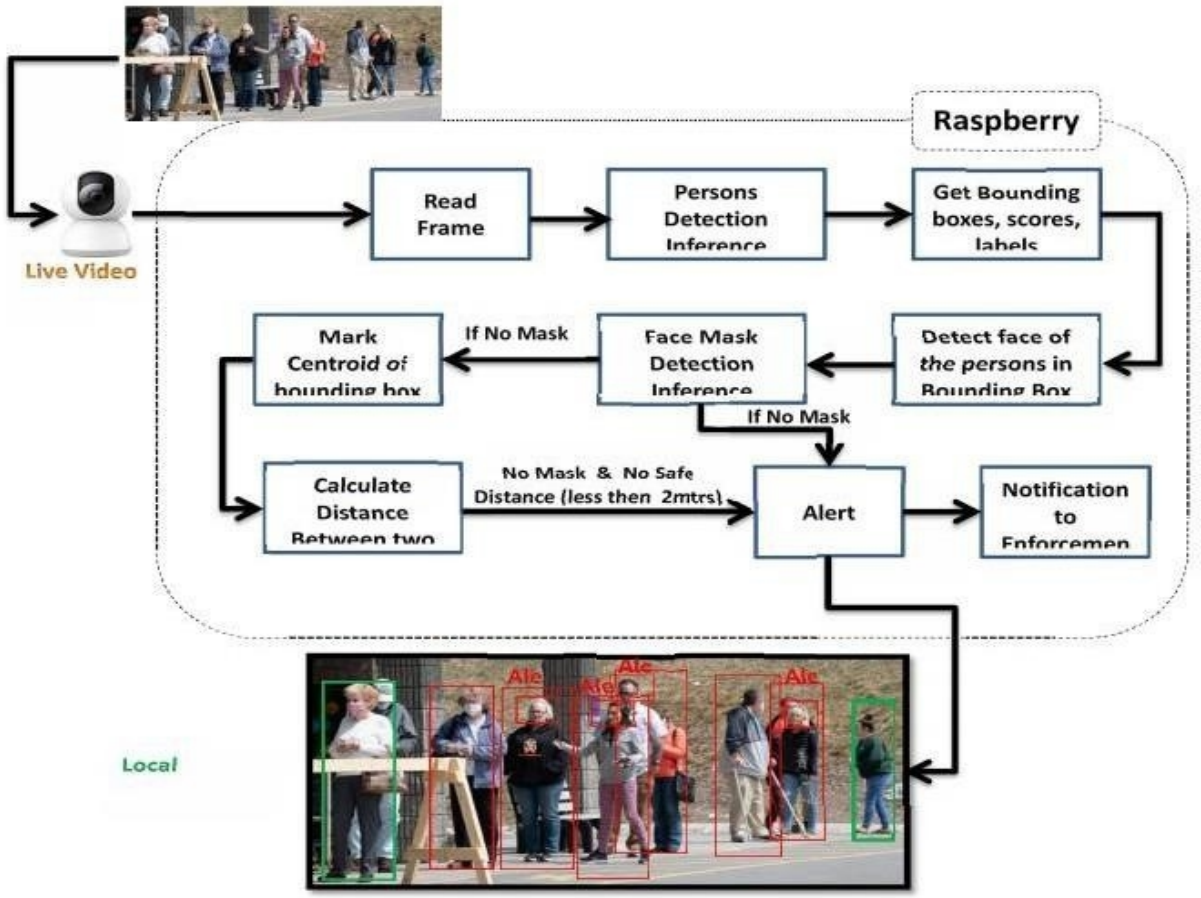

Figure 3.1 Solution architecture of proposed system

The proposed system uses a transfer learning approach to performance optimization with a deep learning algorithm and a computer vision to automatically monitor people in public places with a camera integrated with a raspberry pi4 and to detect people with mask or no mask. We also do fine tuning, which is another form of transfer learning, more powerful than just the feature extraction.

This pandemic has left us no other choice but to wear a mask and maintain social distancing along with sanitation measures until the vaccine is fully developed and distributed. However implementation & regulation of such sanitation measures and personal protection gears like masks, sanitisers demands a lots of management related issues. It's very difficult to monitor people wearing mask or people without mask in large scale , It requires a lot of human resources to constantly monitor a lot of people.

To evaluate mask wearing as a community mitigation strategy, consider performing the following activities:

- **Prioritize and focus** the evaluation on what is feasible, useful, and applicable to the jurisdiction (e.g., state, county, city, organization) and to a community's needs and context.
- **Collect new data or review existing data** on mask-wearing appropriate for the focus and align with other activities that monitor and evaluate COVID-19 community mitigation strategies.
- **Conduct analyses** that compare changes over time and geographic and socio-demographic variations to assess how mask-wearing can reduce COVID-19 transmission, morbidity, and related mortality.
- **Consider each community's cultural context** when interpreting findings and making conclusions.

Use evaluation findings to

- Inform decisions about strengthening, focusing, and relaxing guidance for mask-wearing.
- Monitor disparities and social determinants of health to understand how different

populations participate in, and are affected by, wearing masks, including stigma associated with mask-wearing in specific populations.

- Share lessons learned, along with tools and resources needed to effectively implement mask-wearing as a COVID-19 community mitigation strategy.

This requires a huge management challenge. Our project is aimed to distinguish people from wearing masks and those without wearing so that willful defaulters can be differentiated and accordingly they can be penalized. It uses a deep learning model to constantly learn and distinguish people.

3.2. WORKING MODEL

We are loading the MobileNet V2 with pre-trained ImageNet weights, leaving the network head off and constructing a new FC head, attaching it to the base instead of the old head, and freezing the base layers of the network. The weights of these base layers will not be changed during the fine tuning phase of the backpropagation, while the head layer weights will be adjusted. After data is prepared and the model architecture is set up for fine tuning, then the model is compiled and trained.

A very small learning rate is used during the retraining of the architecture to ensure that the convolutional filters already learned do not deviate dramatically and experiments have been carried out with OpenCV, TensorFlow using Deep Learning and Computer Vision in order to inspect the safe social distance between detected persons and face masks detection in real-time video streams. The main contribution of the proposed system is three components: person detection, safe distance measurement between detected persons, face mask detection. Real-time person detection is done with the help of Single Shot object Detection (SSD) using MobileNet V2 and OpenCV, achieves 91.2% mAP, outperforming the comparable state-of-the-art Faster R-CNN model.

There are two predominant approaches to the face recognition problem: Geometric (feature based) and photometric (view based). As researcher interest in face recognition continued, many different algorithms were developed, three of which have been well studied in face recognition literature. Recognition algorithms can be divided into two main approaches: 1. Geometric: Is based on geometrical relationship between facial landmarks, or in other words

the spatial configuration of facial features.

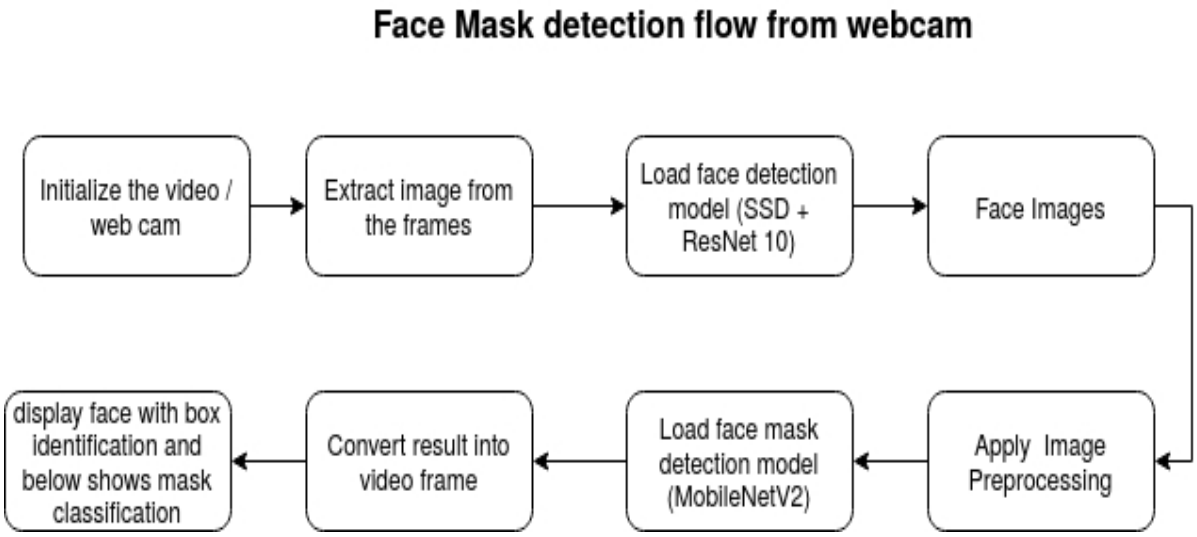

Figure 3.2. Flow Diagram of Face Mask Detection From Webcam

That means that the main geometrical features of the face such as the eyes, nose and mouth are first located and then faces are classified on the basis of various geometrical distances and angles between features. (Figure 3) 2. Photometric stereo: Used to recover the shape of an object from a number of images taken under different lighting conditions.

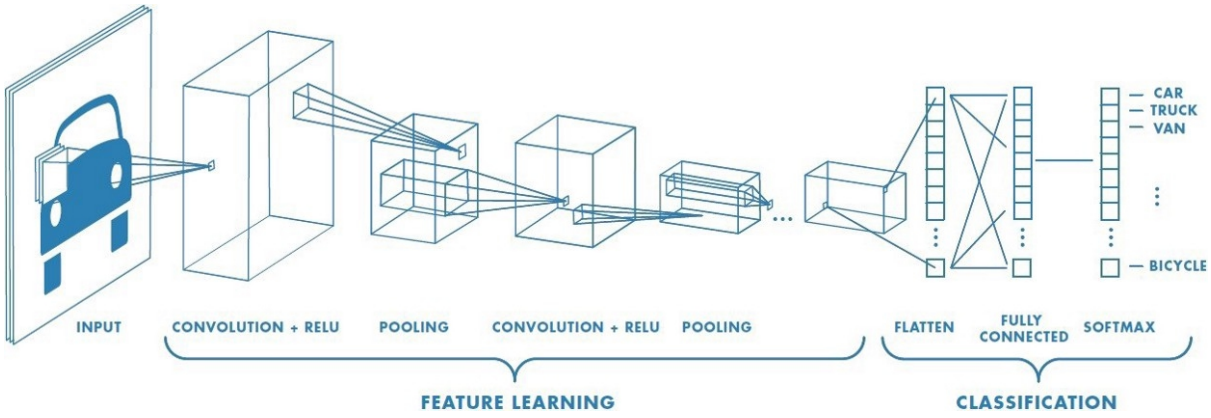

Figure 3.3. Convolutional Neural Network

In the case of a standard feed-forward NN, each input neuron would directly map to a feature in the dataset, and the assumption here is that each neuron (and thus feature) is completely

independent of each other. However, this is not the case for image data. Pixels in an image

have both Spatial and Temporal dependencies — if you imagine an image of the moon in the night sky, you would expect all of the pixels near the moon to have similar pixel values (they should all be around 200–255), and the further away from the moon the pixel is, the darker the pixel becomes (and closer it's value reduces towards 0). A standard feed-forward Neural Network would not be able to preserve this type of Spatial and Temporal information and it's performance is constrained to the information it can gain from each individual pixel in the image without considering other pixels nearby.

The Convolutional Neural Network, on the other hand, can capture these dependencies through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and the reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

CHAPTER 4

EXPERIMENTAL RESULTS

4.1. INTRODUTION

A bounding box will be displayed around every person detected. Although SSD is capable of detecting multiple objects in a frame, it is limited to the detection of a single person in this system. To calculate the distance between two persons first the distance of person from camera is calculated using triangle similarity technique, we calculate perceived focal length of camera, we assumed person distance D from camera and person's actual height H=165cms and with SSD person detection pixel height P of the person is identified using the bounding box coordinates. Using these values, the focal length of the camera can be calculated using the formula below:

$F = (P \times D) / H$

Then we use the real person's height H, the person's pixel height P, and the camera's focal length F to measure the person's distance from the camera. The distance from the camera can be determined using the following:

$D1 = (H \times F) / P$

After calculating the depth of the person in the camera, we calculate the distance between two people in the video. A number of people can be detected in a video. Thus, the Euclidean distance is measured between the mid-point of the bounding boxes of all detected individuals. By doing this, we got x and y values, and these pixel values are converted into centimeters. We have the x, y and z (the person's distance from the camera) coordinates for each person in cms. The Euclidean distance between each person detected is calculated using (x, y, z) coordinates. If the distance between two people is less than 2 meters or 200 centimeters, a red bounding box is shown around them, indicating that they do not maintain a social distance.

In the proposed system transfer learning is used on top of the high performing pre- trained SSD model for face detection with mobileNet V2 architecture as backbone to create a lightweight model that is accurate and computationally efficient, making it easier to deploy the model to raspberry pi. We used custom face crop datasets of about 3165 images annotated in mask and no mask. Annotated images are used to train a deep learning binary classification model that classifies the input image into the mask and no mask categories using the output class confidence.

The result of the SSD model extracts a person mask and displays a bounding box.
The proposed system monitors public places continuously and when a person without a mask is detected his or her face is captured and an alert is sent to the authorities with face image and at the same time the distance between individuals is measured in real time, if more than 20 persons have been identified continuously breaching safe social distance standards at the threshold time, then alert is sent to the control center at the State Police Headquarters to take further action.

This system can be used in real-time applications requiring a secure monitoring of social distance between people and the detection of face masks for safety purposes due to the outbreak of Covid-19. Deploying our model to edge devices for automatic monitoring of public places could reduce the burden of physical monitoring, which is why we choose to use this architecture. This system can be integrated with edge device for use in airports, railway stations, offices, schools and public places to ensure that public safety guidelines are followed.

4.2. RESULTS

Proposed system is a deep learning solution that uses OpenCV and TensorFlow, to train model. We combine the deep learning MobileNetV2 modal with the SSD framework for a fast and efficient deep learning solution for real-time human detection in video streams and use a triangular similarity technique to measure distance between persons detected by camera in real time in public places and comprises customized data collection to resolve a face mask detection model with variance in the types of face masks worn by the public in real time by means of a transfer of learning to a pre-trained SSD face detector.

In the proposed system, four steps are followed, such as:

- Data collection and pre-processing
- Model development and training
- Model testing
- Model implementation

4.2.1. Data Collection and Pre - processing :

The proposed system uses a custom data set consisting of face images with different types of face masks which are labeled and used for the training of our models. We use the existing background subtraction algorithm in a pre-processing step. The real-time automated detection of social distance maintenance and the verification of persons wearing masks or not are performed by the SSD algorithm. The dataset used to train our proposed face mask detector consists of 3165 images. Before the custom face mask image dataset is labelled, the data set is divided into the training data set and the testing data set. The Training data set should consist of 80% images to train the algorithm effectively and for prediction accuracy and the Testing data set should consist of 20% images to test the prediction accuracy of the algorithm. The images in the training data collection are classified into two categories: mask and no mask.

Figure 4.1. Sample Images of custom dataset

4.2.2. Model building and Training:

Our proposed framework uses the transfer learning approach and will fine-tune the MobileNetV2 model, which is a highly efficient architecture that can be applied to edge devices with limited computing power, such as raspberry pi4 to detect people in real time. We used 80% of our total custom data set to train our model with a single shot detector, which takes only one shot to detect multiple objects that are present in an image using multibox. The custom data set is loaded into the project directory and the algorithm is trained on the basis of the labeled images. In pre-processing steps, the image is resized to 224×224 pixels, converted to numpy array format and the corresponding labels are added to the images in the dataset before using our SSD model as input to build our custom model with MobileNetV2 as the backbone and train our model using the.

TensorFlow Object Detection API. Before model training begins, tensorflow helps in Data augmentation and download pre-trained ImageNet weights to make the algorithm's prediction efficiency accurate. After downloading the pre-trained weights and creating a new fully-connected (FC) head, the SSD algorithm is trained with both the pre-trained ImageNet weights and the annotated images in the custom data set by tuning the head layer weights without updating weights of base layers.

We trained our model for 1000 steps using the Adam optimizing algorithm, the learning decay rate for updating network weights, and the binary cross- entropy for mask type

classification. Parameters were initialized for the initial learning rate of INIT_LR = 1e-4, number of epoch EPOCHS = 20 and batch size BS = 32. We used webcam for social distance monitoring using cv2 and after a person has been identified, we start with bounding box coordinates and computing the midpoint between the top-left and the bottom-left along with the top-right and bottom-right points. We measure the Euclidean distance between the points in order to determine the distance between the people in the frame.

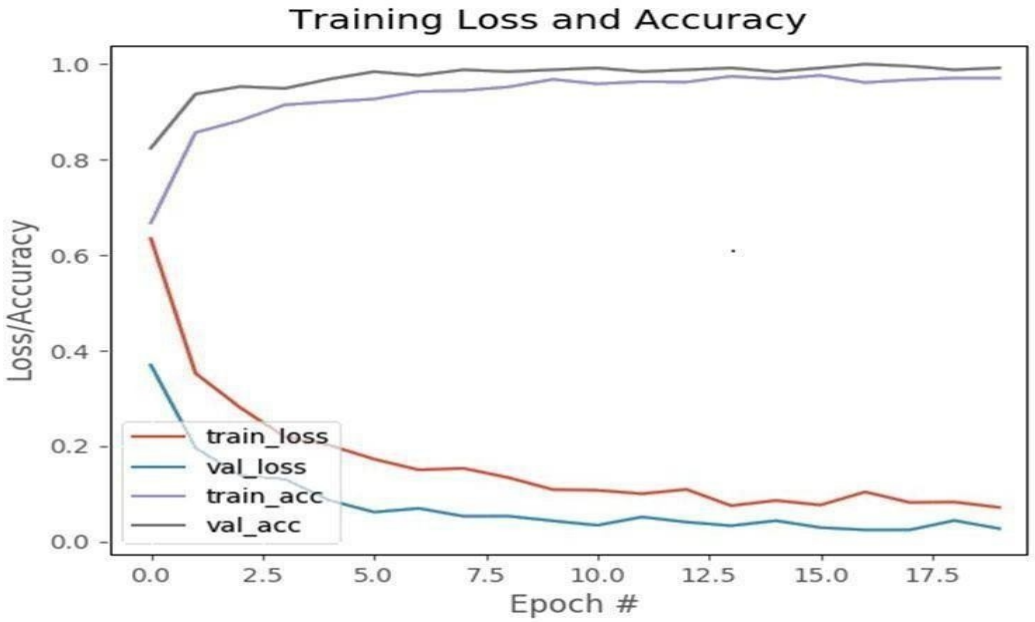

Figure 4.2. Model training accuracy/loss Curves

4.2.3. Model Testing

The proposed system operates in an automated way and helps to automatically perform the social distance inspection process. Once the model is trained with the custom data set and the pre- trained weights given, we check the accuracy of the model on the test dataset by showing the bounding box with the name of the tag and the confidence score at the top of the box. The proposed model first detects all persons in the range of cameras and shows a green bounding box around each person who is far from each other after that model conducts a test on the identification of social distances maintained in a public place, if persons breaching social

distance norms bounding box color changes to red for those persons and simultaneously face mask detection is achieved by showing bounding boxes on the identified person's face with mask or non-mask labeled and also confidence scores. If the mask is not visible in the faces, and if the social distance is not preserved, the system generates a warning and send alert to monitoring authorities with face image.

The system detects the social distancing and masks with a precision score of 91.7% with confidence score 0.7, precision value 0.91 and the recall value 0.91 with FPS = 28.07.

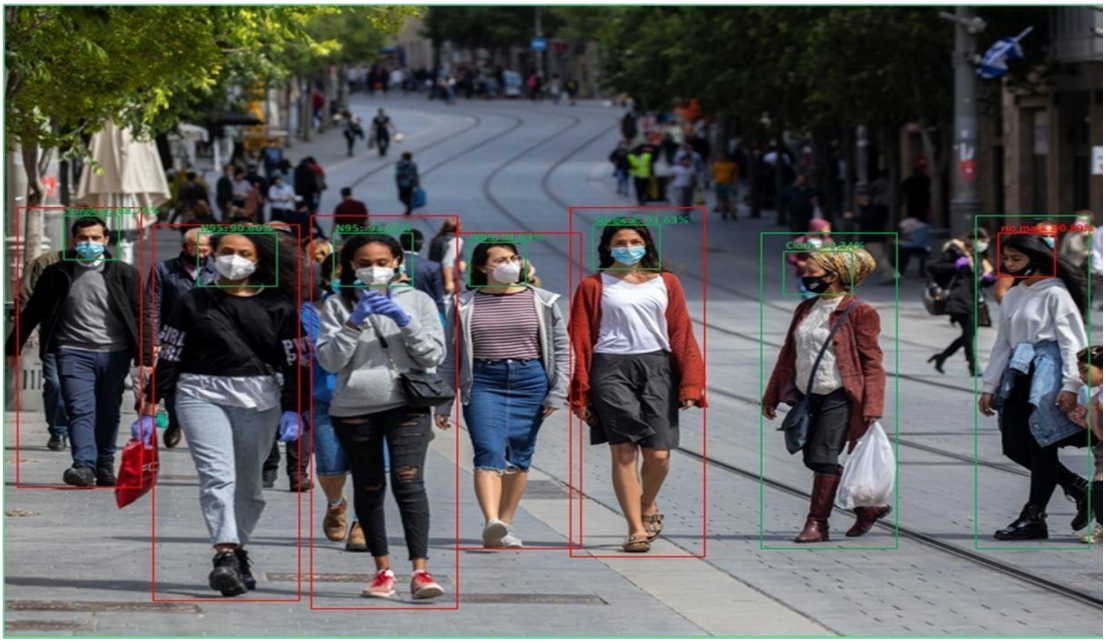

Figure 4.3. Test result of model

Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the feature map. This is primarily to reduce the computational power required to process the data through dimensionality reduction (think back to those 8K images). Furthermore, the other important role of this layer is to extract the dominant features which are both rotational and positional invariant within the input image, thus helping to maintain strong prediction capability,

There are two common types of Pooling: Max Pooling and Average Pooling. **Max Pooling** returns the **maximum value** from the portion of the image covered by the filter. On the other hand, **Average Pooling** returns the **average of all the values** from the portion of the image covered by the filter.

Max Pooling also performs as a **Noise Suppressant**. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction. On the other

hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that **Max Pooling generally performs a lot better than Average Pooling**.

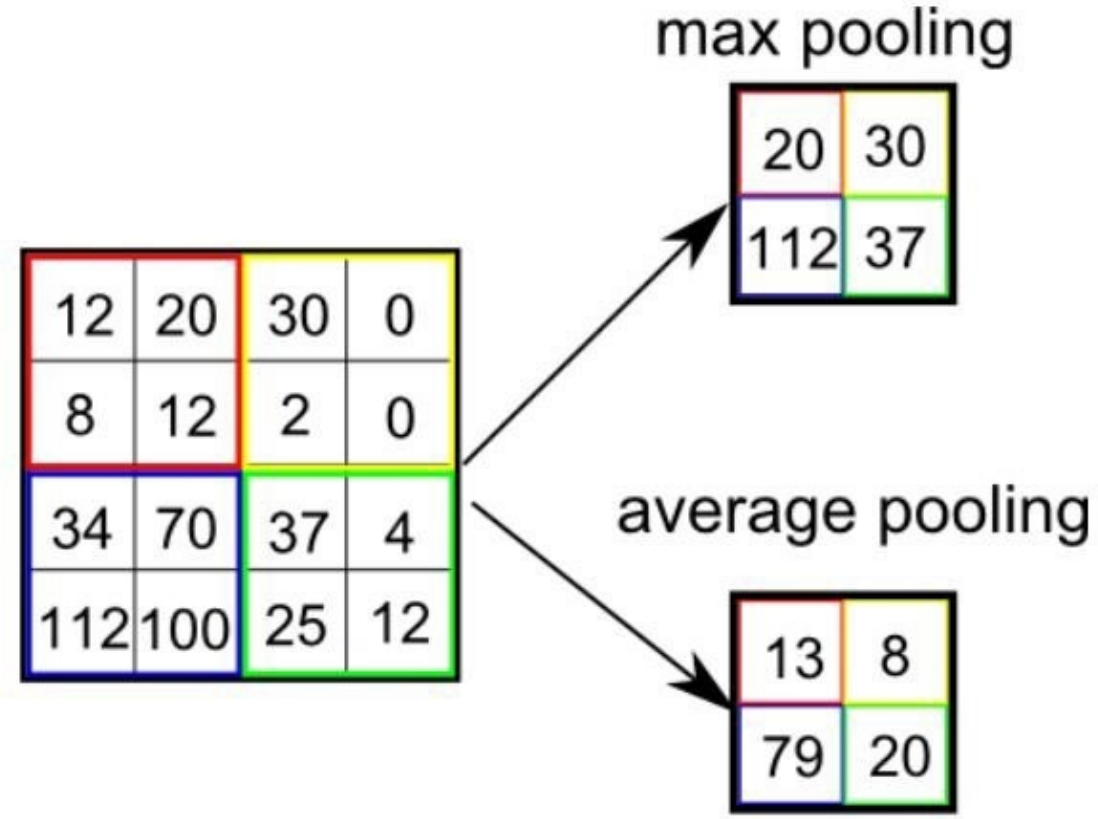

Figure 4.4. Max Pooling and average pooling Difference

Passing through the relevant Convolutional and Pooling layers should enable the model to successfully understand the most important features within your image. It is at this stage where we can flatten the data and pass it through a Feed-forward Neural Network to allow the classification into an output category to take place.

To begin developing the Convolutional Neural Network, we first need to import the required functions from the Keras ML library. The CNN architecture is then defined using a series of the model.add function. The architecture for this project contains:

- 2 Convolutional Layers, both of which have a 3 x 3 convolutional window and ReLU activation function. The first Conv layer produces 200 feature maps, whereas the second produces 100 feature maps. Stride and Padding has not been altered in this project.
- 2 Max Pooling layers, both of which have window sizes of 2 x 2.
- 2 Fully-Connected layers (one hidden layer, one output layer). The hidden layer has 50 neurons, and the output layer has only 2 (one for each class). Softmax activation function has been used to calculate the class probabilities.

The model also uses dropout to help prevent overfitting of the model to the training data. More information on this hyper-parameter can be accessed

Now that the model has been trained and it's performance has been evaluated on the unseen images from the test set, we can think about how we want to feed our live webcam images into the model for prediction, and how we would like the prediction to be fed back to the user in a suitable format. All of the images from the webcam feed will need to undergo the same data pre-processing steps that we carried out on the training data, with the additional step of identifying where a face is located within the webcam image. The visual above shows how the webcam feed will be processed from the original image to a cropped "face only" image, to the fully pre-processed image ready for classification.

In terms of the visual feedback from the classification task, we have decided that the output will come in the form of two rectangular boxes (one will have a Green outline if a mask is present, and a Red outline if a mask is not present) and the other will encompass the Class label ('Mask' or 'No Mask'). These boxes will be shown on the live webcam feed around our cropped "face only" image.

Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the feature map. This is primarily to reduce the computational power required to process the data through dimensionality reduction (think back to those 8K images).

Furthermore, the other important role of this layer is to extract the dominant features which are both rotational and positional invariant within the input image, thus helping to maintain strong prediction capability,

“There are two common types of Pooling: Max Pooling and Average Pooling. **Max Pooling** returns the **maximum value** from the portion of the image covered by the filter. On the other hand, **Average Pooling** returns the **average of all the values** from the portion of the image covered by the filter.

Max Pooling also performs as a **Noise Suppressant**. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that **Max Pooling generally performs a lot better than Average Pooling**.”

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1. CONCLUSION

In this paper, we proposed an approach that uses computer vision and MobileNet V2 architecture to help maintain a secure environment and ensure individuals protection by automatically monitoring public places to avoid the spread of the COVID-19 virus and assist police by minimizing their physical surveillance work in containment zones and public areas where surveillance is required by means of camera feeds with in real-time.

Thus, this proposed system will operate in an efficient manner in the current situation when the lockout is eased and helps to track public places easily in an automated manner. We have addressed in depth the tracking of social distancing and the identification of face masks that help to ensure human health. The implementation of this solution was successfully tested in real-time by deploying model . The solution has the potential to significantly reduce violations by real-time interventions, so the proposed system would improve public safety through saving time and helping to reduce the spread of coronavirus. This solution can be used in places like temples, shopping complex, metro stations, airports, etc.

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 MobileNet 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 adoptweights 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 ofthe model is a continuous process andwe 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 dataset. By the development of face mask detection we can detect if the person is wearing a face mask and allow their entry would be of great help to the society

5.2. FUTURE SCOPE

The above mentioned use cases are only some of the many features that were incorporated as part of this solution. We assume there are several other cases of usage that can be included in this solution to offer a more detailed sense of safety. Several of the currently under development features are listed below in brief:

Coughing and Sneezing Detection*:*

Chronic coughing and sneezing is one of the key symptoms of COVID-19 infection as per WHO guidelines and also one of the major route of disease spread to non-infected public. Deep learning based approach can be proved handy here to detect & limit the disease spread by enhancing our proposed solution with body gesture analysis to understand if an individual is coughing and sneezing in public places while breaching facial mask and social distancing guidelines and based on outcome enforcement agencies can be alerted.

Temperature detection:

Elevated body temperature is another key symptom of COVID-19 infection, at present scenario thermal screening is done using handheld contact less IR thermometers where health worker need to come in close proximity with the person need to be screened which makes the health workers vulnerable to get infected and also its practically impossible to capture temperature for each and every person in public places, the proposed use-case can be equipped with thermal cameras based screening to analyze body temperature of the peoples in public places that can add another helping hand to enforcement agencies to tackle the pandemic effectively.
PDM (Point Distribution Model):

Independently of computerized image analysis, and before ASMs were developed, researchers developed statistical models of shape . The idea is that once you represent shapes as vectors, you can apply standard statistical methods to them just like any other multivariate object. These models learn allowable constellations of shape points from training examples

and use principal components to build what is called a Point Distribution Model.

These have been used in diverse ways, for example for categorizing Iron Age broaches.Ideal Point Distribution Models can only deform in ways that are characteristic of the object. Cootes and his colleagues were seeking models which do exactly that so if a beard, say, covers the chin, the shape model can \override the image" to approximate the position of the chin under the beard. It was therefore natural (but perhaps only in retrospect) to adopt Point Distribution Models. This synthesis of ideas from image processing and statistical shape modelling led to the Active Shape Model.The first parametric statistical shape model for image analysis based on principal components of inter-landmark distances was presented by Cootes and Taylor in. On this approach, Cootes, Taylor, and their colleagues, then released a series of papers that cumulated in what we call the classical Active Shape Model.

Low Level Analysis

Based on low level visual features like color, intensity, edges, motion etc.Skin Color Base Color is a vital feature of human faces. Using skin-color as a feature for tracking a face has several advantages. Color processing is much faster than processing other facial features. Under certain lighting conditions, color is orientation invariant. This property makes motion estimation much easier because only a translation model is needed for motion estimation.Tracking human faces using color as a feature has several problems like the color representation of a face obtained by a camera is influenced by many factors (ambient light, object movement, etc).

REFERENCE

1. S. Wang Chen, Horby Peter W, Hayden Frederick G, Gao George F. A novel coronavirus epidemic of global concern for health. It's the Lancet. 2020;395(10223):470-473. 10.1016 / S0140-6736(20)30185-9.
2. Matrajt L, Leung T. Evaluating the efficacy of social distancing strategies to postpone or flatten the curve of coronavirus disease. Emerg Infect Dis, man. 2020:
3. Chen, S., Zhang, C., Dong, M., Le, J., R., M., 2017b. Using rankingcnn for age estimates, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
4. Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, "Mobilenetv2: Inverted residues and linear bottlenecks," IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510-4520. 2018.
5. C.Fu, W.Liu, A.Ranga, A. Tyagi, A. Berg, "DSSD: deconvolutional single shot detector model," arXiv preprint arXiv:1701.06659, (2017)
6. Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, "Type Pyramid Networks for Object Detection," IEEE Conference Proceedings on Computer Vision and Pattern Recognition, pp. 2117-2125. 2017.
7. GOODFELLOW, I., BENGIO, Y., & COURVILLE, A. (2016). Deep learning process. Chapter 6.
8. S. S. Farfade, M. J. Saberian, and L. Li. Multi-view face recognition using deep convolutional neural networks. In ACM ICMR, pages 643–650, 2015
9. Masita, K. L., Hasan, A. N., and Satyakama, P., 2018. Pedestrian identification by mean of R- CNN object detector. IEEE Latin American Conference on Computational Intelligence (Nov. 2018). DOI=10.1109/LA-CCI.2018.8625210.
10. R. Girshick, “Fast R-CNN,” in Proc. IEEE International Confrence Computer Vision, Dec.2015, pp. 1440–1448.
11. S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with area proposal networks,” in Proc. Adv. Neural Inf. Process. Syst., 2015, pp. 91–99.
12. Liu, H., Chen, Z., Li, Z., and Hu, W. 2018. An powerful method of pedestrian

detection based on YOLOv2. Mathematical Engineering Issues(Dec. 2018). DOI=https://doi.org/10.1155/2018/3518959.

13. P. Viola and M. Jones, "Fast object detection using an enhanced cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA, 2001, pp. I-I.
14. X. Zhu and D. Ramanan. Face detection, pose estimation,and landmark position in the wild.In IEEE CVPR, pages 2879–2886, 2012

15. Howard, Andrew & Zhu, Menglong & Chen, Bo & Kalenichenko, Dmitry & Wang, Weijun & Weyand, Tobias & Andreetto, Marco & Adam, Hartwig. (2017). MobileNets: Effective Convolutional Neural Networks for Mobile Vision Applications.
16. Zhu, X. 2006. Semi-supervised study of learning literature survey. Computer Science, University of Wisconsin-Madison 2(3):4.