Face Mask Detection with Deep Learning

A Project Work Report

Submitted in the partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE WITH SPECIALIZATION IN

BIG DATA ANALYTICS

Submitted by:

Tarun Mahajan 19BCS3805

Under the Supervision of:

Mr. Pulkit Dwivedi

CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,

PUNJAB

May, 2023

BONAFIDE CERTIFICATE

Certified that this project report **"FACE MASK DETECTION USING DEEP LEARNING"** is the bonafide work of "**TARUN MAHAJAN** who carried out the project work under my/our supervision.

SIGNATURE **SIGNATURE**

Dr. AMAN KAUSHIK
HEAD OF DEPARTMENT

Mr. PULKIT DWIVEDI
SUPERVISOR

Submitted for the project viva-voce examination held on
___18/05/23_____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

This research paper “FACE MASK DETECTION USING DEEP LEARNING ”, is a result of interest, dedication and hard work. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to our project mentor, MR. PULKIT DWIVEDI for his guidance and constant supervision as well as for providing necessary information regarding the project & also for her support in completing the project.

I would like to express my gratitude towards my parents and seniors for their kind co-operation and encouragement which help me in completion of this project. My thanks and appreciations also go to my colleagues in developing the project and people who have willingly helped me out with their abilities.

Thanks again to all associated with the project.

-Tarun Mahajan

Table of Contents

List of Figures

List of Tables

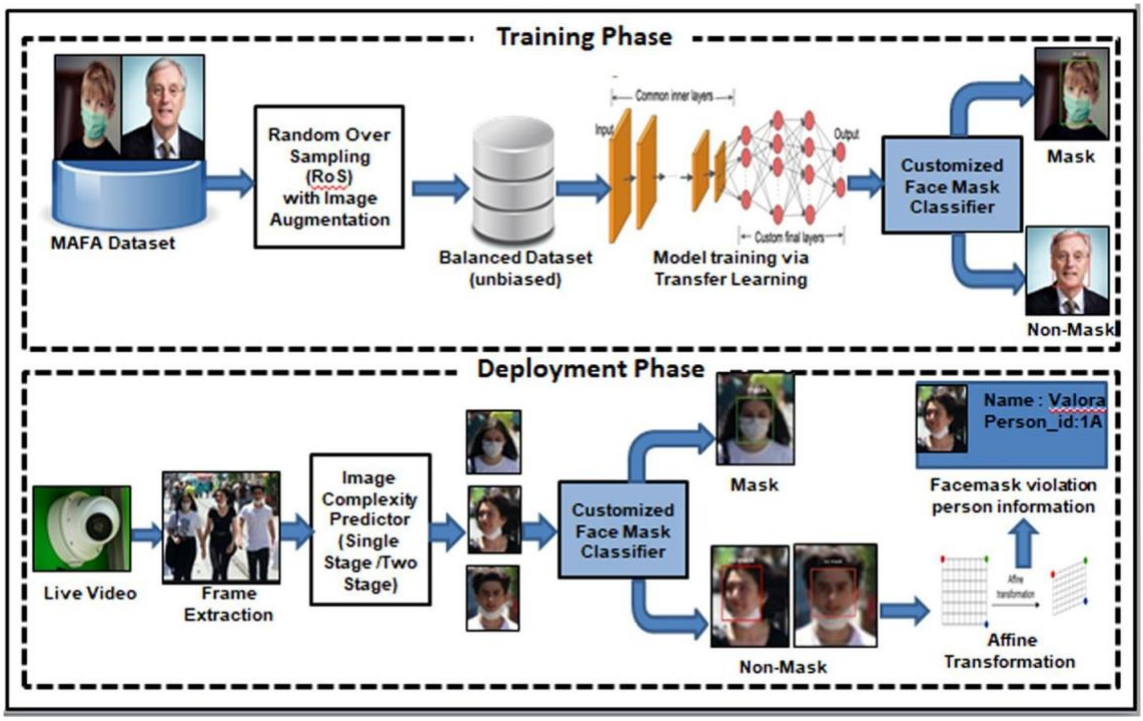
Abstract

Effective strategies to contain the COVID-19 pandemic must focus on mitigating negative impacts on public health and the global economy, and the full picture is yet to be revealed. In the absence of effective antiviral drugs and limited medical resources, WHO recommends multiple measures to control the rate of infection and avoid draining limited medical resources. Mask-wearing is among the non-pharmaceutical interventions that can be used to cut off the main source of SARS-CoV2 droplets expelled by infected people. Regardless of medical resources and discussions of mask diversity, all countries need masks to cover the nose and mouth. To promote community health, this article aims to design a highly accurate and real-time technique that can effectively detect unmasked faces in public to enforce mask-wearing.The proposed technique is a combination of primary and secondary detectors to achieve low temporal drift and high accuracy. We use a large dataset of 45,000 images covering a variety of face images, including faces with masks, faces without masks, faces with and without masks in a faces image, and faces without masks confused image . Training with such a large data set improves the efficiency and accuracy of our technique. We start with ResNet50 as a reference and apply the concept of transfer learning to incorporate high-level semantic information into multiple feature maps. Moreover, we propose to train bounding boxes to improve localization performance during mask detection.The experiment was carried out with three popular basic models, viz. ResNet50, AlexNet and MobileNet. We are studying the possibility of creating plugins for these models using the proposed model, so that very precise results can be obtained in a shorter inference time. We find that the proposed technique achieves high accuracy (98.2%) when implementation is done with ResNet50. Additionally, the proposed model yields 11.07% and 6. Within hours, mask detection achieved 44% improvements in accuracy and recall

over a recently released public baseline model called the RetinaFaceMask detector. The excellent performance of the proposed model is well suited to CCTV installations.

Keyword: Face mask detection, Transfer learning, Object deletion, One-stage detector, Two-stage detector.

Graphical Abstract



CHAPTER - 1

1. INTRODUCTION

1.1 Problem Definition

The 209th World Health Organization (WHO) report from August 16, 2020 reports that coronavirus disease (COVID-19) caused by acute respiratory syndrome (SARS-CoV2) has infected more than 6 million people in the world, killing more than 379,941 people globally[1]. According to Carissa F. Etienne, director of the Pan American Health Organization (PAHO)[2], the keys to controlling the COVID-19 pandemic are social distancing, improving surveillance and strengthening health systems. A recent study by researchers at the University of Edinburgh on understanding measures to respond to the COVID-19 pandemic showed that wearing a face mask or other method of covering the nose and the mouth can avoid the risk of transmission of the coronavirus distances ahead, More than 90% of the distance is travelled by a person's exhaled breath . An extensive study was also conducted to calculate the impact of mask use by the general public on community-wide populations, some of which may have been asymptomatically contagious in New York and Washington State. - tons. The results indicate that near-universal (80%) adoption of even weak masks (20% effective) could prevent 17-45% of predicted deaths and reduce peak daily mortality within two months by 34-58%. Their findings urge the public to use Face masks to limit the spread of the coronavirus. Additionally, as the country reopens after the COVID-19 lockdown, the government and public health agencies are recommending masks as a

necessary measure to protect us when we go out in public. In order to force people to wear masks, it is necessary to develop technology to force people to wear masks before being exposed in public places. Mask detection refers to detecting whether a person is wearing a mask. In fact, the problem is the reverse engineering of face detection, where human faces are detected using different learning algorithms automatically or for security, authentication and monitoring purposes. Face detection is an area of interest in the fields of computer vision and pattern recognition.

1.2 Problem Overview

In the past, many researches have provided complex algorithms for face detection. Preliminary research on face detection was completed in 2001, designing process features and applying traditional machine learning algorithms to train efficient classifiers for detection and recognition [3,4]. Problems encountered with this approach include high design complexity and low detection accuracy. In recent years, face detection methods based on deep convolution neural networks (CNN) have been widely developed [5-8] to improve the detection performance. Although many researchers have worked hard to design efficient face detection and recognition algorithms, there are significant differences between "detecting faces under masks" and, "detecting mask over face". According to the existing literature, there are very few studies trying to detect masks on faces. Therefore, our work aims to develop technology that can accurately detect face masks in public spaces (such as airports, train stations, crowded markets, bus stops, etc.) to limit the spread of the virus. coronavirus, thus contributing to public health care. Additionally, detecting faces with/without masks in public is not easy because the datasets available for detecting human

face masks are relatively small, making model training difficult. Therefore, here we use the concept of transfer learning to transfer the learned cores, which are networks trained for similar face detection tasks on the extended data set. The dataset covers a variety of face images, including faces with masks, faces without masks, faces with and without masks in one image, and aliased images without masks. Using the large dataset of 45,000 images, our technique achieves an excellent accuracy of 98.2%. The most important contributions of the proposed work are:

1. Develop a novel object detection method that combines single-stage and two-stage detectors to accurately locate objects times in real time from video streams using back-end transfer learning.

2. Development of an improved affine transformation to segment facial regions with face size, orientation, and background differences from unsupervised real-time images. This step made it possible to better locate the people who violated the standards for wearing a mask in public spaces/offices.

3. Creating an unbiased mask dataset with an imbalance ratio of almost one.

4. The proposed model requires less memory, which facilitates its deployment in embedded devices for monitoring purposes.

1.3 Hardware Specification

1. Computer with preferably intel i7, 2.5 Ghz octa-core processer and 8 GB RAM and 1 TB of secondary storage.

2. Video cam with preferably 780p- 1080p resolution.

1.4 Software Specification

1. Windows operating system (windows 10 and above).
2. Python 3.7 in an IDE environment.

CHAPTER-2

LITERATURE SURVEY

2.1 Existing System

Computer vision techniques have become increasingly important in recent years, particularly in the area of object recognition and pattern learning. One of the critical applications of computer vision in the current pandemic situation is detecting whether people are wearing masks in public places to curb the spread of the coronavirus.

Object recognition is a critical component of computer vision that encompasses both image classification and object detection. In the case of mask detection, the task involves recognizing the presence of a mask on a person's face in a public area using efficient object recognition algorithms through surveillance devices.

The object recognition pipeline consists of generating region proposals followed by classification of each proposal into the relevant class. There have been significant developments in region proposal techniques, with single-stage and two-stage detectors being widely used. Single-stage detectors detect objects using a single feed-forward convolutional neural network, whereas two-stage detectors detect objects using region proposal networks and subsequent classification.

There are general techniques for improving region proposal detection, including data augmentation and model ensembling, which can be used to improve the accuracy and speed of the object recognition pipeline. In addition, pre-trained models based on these techniques have been developed to improve object recognition performance.

While face detection has been an active area of research for many years, few studies have attempted to detect masks on faces. The development of deep convolutional neural networks has revolutionized the performance of face detection methods. However, detecting masks on faces poses a unique set of challenges due to the varying degrees of occlusion and shape distortion caused by masks.

Despite these challenges, efficient object recognition algorithms can be deployed through surveillance equipment to identify masks on faces in public spaces. The integration of advanced region proposal techniques, such as single-stage and two-stage detectors, with pre-trained models can significantly improve the accuracy and speed of mask detection.

In conclusion, the recent developments in object recognition and region proposal techniques have opened up new possibilities for detecting masks on faces in public places to limit the community spread of the coronavirus. The integration of these techniques with surveillance devices can lead to the development of highly accurate and efficient systems that can play a critical role in ensuring public safety and well-being.

2.1.1 Single-stage detectors

Single-stage detectors treat proposition detection as a simple regression problem by taking an input image and learning the class probabilities and bounding box coordinates. OverFeat[9] and DeepMultiBox[10] are prime examples. YOLO (You Only Look Once)generalizes single-stage methods by demonstrating real-time predictions and achieving significant detection speed, but location accuracy is lower than that of two-stage detectors;is particularly taken into account when the object is small . Basically, the YOLO network divides the image into grids of GxG size, and each grid generates N predictions of bounding boxes. Each bounding box is limited to a single class during prediction, which in turn limits the network to find smaller objects. Additionally, the YOLO network is enhanced to YOLOv2, which includes batch normalization, a high-resolution classifier, and anchor boxes. Additionally, the development of YOLOv3 builds on YOLOv2 by adding an improved backend classifier, multi-sell prediction, and a new network for feature extraction. Although YOLOv3 performed faster than the Single- Shot De-(SSD), it performed poorly in classification accuracy . Additionally, YOLOv3 requires a lot of computing power for inference, so it is not suitable for embedded or mobile devices. Second, due to small convolutional filters, multiple feature maps and multi-scale predictions, SSD network has better performance than YOLO. The main difference between the two architectures is that YOLO uses two fully connected layers, while SSD array uses convolutional layers of different sizes. In addition, RetinaNet[11] pre-proposed by Lin is also a single-stage object detector, which uses image pyramid and loss of focus to detect dense objects in multilayer images and achieves remarkable accuracy and speed Speed is comparable to two-stage detectors.

OverFeat is an integrated framework for using Convolutional Networks for classification, localization and detection. We show how a multiscale and sliding window approach can be efficiently implemented within a ConvNet. We also introduce a novel deep learning approach to localization by learning to predict object boundaries. Bounding boxes are then accumulated rather than suppressed in order to increase detection confidence. We show that different tasks can be learned simultaneously using a single shared network. This integrated framework is the winner of the localization task of the ImageNet Large Scale Visual Recognition Challenge 2013 (ILSVRC2013) and obtained very competitive results for the detection and classifications tasks. In post-competition work, we establish a new state of the art for the detection task. Finally, we release a feature extractor from our best model called OverFeat.

The YOLO architecture is designed to be an end-to-end object detection system, which means it takes in an entire image as input and produces bounding box predictions and class probabilities for all objects in the image as output. The output tensor has a shape of (7, 7, 30), where the first two dimensions represent the spatial location of the cell in the grid and the last dimension contains the predicted information for each cell. Specifically, each cell predicts 2 bounding boxes, 20 class probabilities, and 2 objectness scores.

The objectness score is a measure of how confident the model is that there is an object in the bounding box. The class probabilities represent the probability of the object belonging to each of the 20 classes in the dataset, such as person, car, or dog. The bounding box predictions consist of 4 values: x, y, width, and height, which represent the coordinates of the top-left corner of the bounding box and its width and height, respectively.

To improve the accuracy of YOLO, YOLOv2 was introduced with some modifications. YOLOv2 uses batch normalization to improve the training speed and regularization, and introduces anchor boxes to improve the bounding box prediction accuracy. The anchor boxes serve as reference boxes of different scales and aspect ratios that the model can adjust to fit the object more accurately. YOLOv2 also uses a new network architecture called Darknet-19, which has fewer layers than the original YOLO but achieves higher accuracy.

Despite the improvements made in YOLOv2, there are still some limitations to the YOLO architecture, such as difficulty in detecting small objects and objects that are close together. YOLOv3 was subsequently introduced to address some of these limitations, which uses a feature pyramid network and a new detection head to improve the detection accuracy for small objects and objects at different scales.

SSD, or Single Shot Multibox Detector, is another popular one-stage object detection model that differs from YOLO in its approach to feature extraction. Instead of using a fully convolutional network as in YOLO, SSD employs a series of small convolutional filters at multiple feature maps of different resolutions to capture objects of various sizes. This approach enables SSD to detect objects with a wide range of scales and aspect ratios, while also maintaining high accuracy.

Compared to YOLO, SSD has been found to have lower localization errors and higher recall, especially for smaller objects. However, its accuracy may still lag behind region-based detectors such as Faster R-CNN, particularly for larger objects and complex scenes.

YOLOv2 is an improved version of YOLO that aims to address some of its shortcomings and achieve higher accuracy while maintaining real-time processing speed. One key improvement is the

use of anchor boxes, which allows the network to predict multiple bounding boxes of different aspect ratios and scales for each object, leading to more precise localization. Additionally, YOLOv2 uses batch normalization and residual connections to improve feature extraction and reduce overfitting.

The training process of YOLOv2 involves pre-training a classification network on large-scale image classification datasets like ImageNet, then fine-tuning it for object detection with larger input image sizes. By using larger input images and fewer training epochs, YOLOv2 can achieve higher accuracy than the original YOLO while still maintaining real-time processing speed.

Overall, both RetinaNet, SSD, and YOLO (including YOLOv2) are popular one-stage object detection models that offer a good balance between accuracy and real-time processing speed. Choosing the most suitable model depends on the specific requirements of the application, such as the size and complexity of the objects to be detected, the available computational resources, and the desired trade-off between speed and accuracy.

In 2017, Joseph Redmon (a Graduate Student at the University of Washington) and Ali Farhadi (a PRIOR team lead at the Allen Institute for AI) published the YOLO9000: Better, Faster, Stronger paper at the CVPR conference. The authors proposed two state-of-the-art YOLO variants in this paper: YOLOv2 and YOLO9000; both were identical but differed in training strategy.

YOLOv2 was trained on standard detection datasets like PASCAL VOC and MS COCO. At the same time, the YOLO9000 was designed to predict more than 9000 different object categories by jointly training it on the MS COCO and ImageNet datasets.

RetinaNet is a significant advancement in one-stage object detection models due to its ability to address class imbalance and effectively detect objects in dense scenes. The use of a focal loss function during training helps to focus on hard negative examples, which can significantly improve the detection accuracy of the model. Additionally, RetinaNet's unified architecture, consisting of a backbone network and two task-specific subnetworks, makes it a simple and efficient model for object detection.

The backbone network of RetinaNet is responsible for computing a convolutional feature map over the input image. The backbone network is usually a pre-trained convolutional neural network, such as ResNet, which is fine-tuned during training. The first task-specific subnet of RetinaNet performs convolutional object classification on the feature map generated by the backbone network. This subnet assigns a class label to each object proposal generated by the model.

The second task-specific subnet performs convolutional bounding box regression, which predicts the coordinates of the bounding boxes surrounding the objects in the image. The bounding box coordinates are predicted relative to a set of pre-defined anchor boxes, which are used as a reference for the regression task.

RetinaNet's architecture is specifically designed for one-stage, dense detection, which is the task of detecting objects in dense scenes where objects overlap or occlude each other. The model achieves this by incorporating feature pyramids into the backbone network and by using anchor boxes with different scales and aspect ratios. This approach allows RetinaNet to detect objects of different sizes and shapes in dense scenes accurately.

In summary, RetinaNet is a powerful one-stage object detection model that addresses class imbalance during training through the use of focal loss. Its unified architecture, consisting of a backbone network and two task-specific subnetworks, makes it a simple and efficient model for object detection. RetinaNet's ability to handle dense scenes makes it a valuable tool for detecting objects in real-world scenarios.

We can see the motivation for focal loss by comparing with two-stage object detectors. Here class imbalance is addressed by a two-stage cascade and sampling heuristics. The proposal stage (e.g., Selective Search, EdgeBoxes, DeepMask, RPN) rapidly narrows down the number of candidate object locations to a small number (e.g., 1-2k), filtering out most background samples. In the second classification stage, sampling heuristics, such as a fixed foreground-to-background ratio, or online hard example mining (OHEM), are performed to maintain a manageable balance between foreground and background.

In contrast, a one-stage detector must process a much larger set of candidate object locations regularly sampled across an image. To tackle this, RetinaNet uses a focal loss function, a dynamically scaled cross entropy loss, where the scaling factor decays to zero as confidence in the correct class increases. Intuitively, this scaling factor can automatically down-weight the contribution of easy examples during training and rapidly focus the model on hard examples.

2.1.2. Two-Stage Detectors

Compared to single-stage detectors, two-stage detectors follow long lines of inference in computer vision to predict and classify region proposals. They first predict the propositions in the images, then apply classifiers to those regions to classify the potential detections.Various two-step region proposal models have been proposed by researchers in the past. Regional convolutional neural networks, also simply called R-CNN[12] , were described by Ross Girshick et al., 2014. This is probably the first large-scale application of CNN to the problem of localization and object recognition. The model has been successfully demonstrated times on benchmark datasets such as VOC-2012 and ILSVRC-2013 and has produced state-of-the-art results. Basically, R-CNN applies a selective search algorithm to retrieve a set of object proposals in an initial step, and applies a Support Vector Machine (SVM) classifier in a later step to predict objects and related classes. Spatial Pyramid Pooling SPPNet[13] (modified R-CNN with SPP layers) collects features from multiple region proposals and is fed into a fully connected layer for classification. The ability of SPNN to compute the feature map of the entire image in a single shot leads to a significant speed-up ofobject detection, which is nearly 20 times that of R-CNN. Then, Fast R-CNN is an extension of R-CNN and SPPNet .It introduces a new layer called Region of Interest (RoI) clustering layer between the shared convolutional layers to refine the model. Moreover, it allows to train the detector and the regressor simultaneously without changing the configuration of the network. Although Fast-

R-CNN effectively integrates the advantages of R-CNN and SPPNet, it still lacks the detection speed of compared to single-phase detectors.

Region-based fully convolutional networks (R-FCN) is another two-stage object detection model that aims to address the computational redundancy issue of Faster R-CNN. Instead of using RoI pooling to extract features from each region proposal, R-FCN applies position-sensitive score maps and position-sensitive RoI pooling to directly produce class scores and bounding box regressions. This reduces the amount of computation required to extract features for each region proposal and improves the detection speed.

R-FCN also allows for full training and inference in backpropagation, which means that the entire network can be trained end-to-end. This is in contrast to Faster R-CNN, where the RPN and Fast R-CNN stages are trained separately and then combined during inference. End-to-end training allows R-FCN to better optimize all components of the network for object detection.

Overall, R-FCN achieves state-of-the-art accuracy on object detection benchmarks like PASCAL VOC and COCO, while also maintaining a fast detection speed.

Additionally, Faster R-CNN is a hybrid of Fast R-CNN and Regional Proposal Network (RPN). It achieves near-free region proposals by progressively integrating individual blocks of a object detection system (e.g. proposal detection, feature extraction, and bounding box regression) in a single step .Although this merge breaks the speed bottleneck of Fast R-CNN, has computational redundancy in the next detection step. Region-based fully convolutional networks (R-FCN) are the only models that allow full training and inference in backpropagation .

Feature pyramid nets (FPNs) can detect non-uniform objects, but are the least used by researchers due to high computational cost and increased memory usage . Additionally, Mask R-CNN augments Faster. R-CNN by including segmented mask predictions on each RoI . Although two-step object detection gives high accuracy, it is limited by the low real-time inference speed of CCTV . Further, Faster R-CNN is an amalgam of fast R-CNN and Region Proposal Network (RPN). It enables nearly cost-free region proposals by gradually integrating individual blocks (e.g. proposal detection, feature extraction and bounding box regression) of the object detection system in a single step [20], [21]. Although this integration leads to the accomplishment of break-through for the speed bottleneck of Fast R-CNN but there exists a computation redundancy at the subsequent detection stage. The Region-based Fully Convolutional Network (R-FCN) is the only model that allows complete backpropagation for training and inference [22], [23]. Feature Pyramid Networks (FPN) can detect non-uniform objects, but least used by researchers due to high computation cost and more memory usage [24]. Furthermore, Mask R-CNN strengthens Faster R-CNN by including the prediction of segmented masks on each RoI [25]. Although two-stage yields high object detection accuracy, but it is limited by low inference speed in real-time for video surveillance [14].

The R-CNN detector first generates region proposals using an algorithm such as Edge Boxes[1]. The proposal regions are cropped out of the image and resized. Then, the CNN classifies the cropped and resized regions. Finally, the region proposal bounding boxes are refined by a support vector machine (SVM) that is trained using CNN features. Use the train RCNN Object Detector function to train an R-CNN object detector. The function returns an rcnn Object Detector object that detects objects in an image.

As in the R-CNN detector , the Fast R-CNN detector also uses an algorithm like Edge Boxes to generate region proposals. Unlike the R-CNN detector, which crops and resizes region proposals, the Fast R-CNN detector processes the entire image. Whereas an R-CNN detector must classify each region, Fast R-CNN

pools CNN features corresponding to each region proposal. Fast R-CNN is more efficient than R-CNN, because in the Fast R-CNN detector, the computations for overlapping regions are shared.

Use the trainFastRCNNObjectDetector function to train a Fast R-CNN object detector. The function returns a fastRCNNObjectDetector that detects objects from an image.

The Faster R-CNN[4] detector adds a region proposal network (RPN) to generate region proposals directly in the network instead of using an external algorithm like Edge Boxes. The RPN uses Anchor Boxes for Object Detection. Generating region proposals in the network is faster and better tuned to your data.

Use the trainFasterRCNNObjectDetector function to train a Faster R-CNN object detector. The function returns a fasterRCNNObjectDetector that detects objects from an image.

This family of object detectors uses region proposals to detect objects within images. The number of proposed regions dictates the time it takes to detect objects in an image. The Fast R-CNN and Faster R-CNN detectors are designed to improve detection performance with a large number of regions.

SPP "spatial pyramid pooling", to eliminate the above requirement. The new network structure, called SPP-net, can generate a fixed-length representation regardless of image size/scale. Pyramid pooling is also robust to object deformations. With these advantages, SPP-net should in general improve all CNN-based image classification methods. On the ImageNet 2012 dataset, we demonstrate that SPP-net boosts the accuracy of a variety of CNN architectures despite their different designs. On the Pascal VOC 2007 and Caltech101 datasets, SPP-net achieves state-of-the-art classification results using a single full-image representation and no fine-tuning.

The power of SPP-net is also significant in object detection. Using SPP-net, we compute the feature maps from the entire image only once, and then pool features in arbitrary regions (sub-images) to generate fixed-length representations for training the detectors. This method avoids repeatedly computing the

convolutional features. In processing test images, our method is 24-102x faster than the R-CNN method, while achieving better or comparable accuracy on Pascal VOC 2007. SPP-Net is a convolutional neural architecture that employs spatial pyramid pooling to remove the fixed-size constraint of the network. Specifically, we add an SPP layer on top of the last convolutional layer. The SPP layer pools the features and generates fixed-length outputs, which are then fed into the fully-connected layers (or other classifiers). In other words, we perform some information aggregation at a deeper stage of the network hierarchy (between convolutional layers and fully-connected layers) to avoid the need for cropping or warping at the beginning.

Detecting objects in different scales is challenging in particular for small objects. We can use a pyramid of the same image at different scale to detect objects (the left diagram below). However, processing multiple scale images is time consuming and the memory demand is too high to be trained end-to-end simultaneously. Hence, we may only use it in inference to push accuracy as high as possible, in particular for competitions, when speed is not a concern. Alternatively, we create a pyramid of feature and use them for object detection (the right diagram). However, feature maps closer to the image layer composed of low-level structures that are not effective for accurate object detection.

Feature Pyramid Network (**FPN**) is a feature extractor designed for such pyramid concept with accuracy and speed in mind. It replaces the feature extractor of detectors like Faster R-CNN and generates multiple feature map layers (**multi-scale feature maps**) with better quality information than the regular feature pyramid for object detection.

Detecting objects in different scales is challenging in particular for small objects. We can use a pyramid of the same image at different scale to detect objects (the left diagram below). However, processing multiple

scale images is time consuming and the memory demand is too high to be trained end-to-end simultaneously. Hence, we may only use it in inference to push accuracy as high as possible, in particular for competitions, when speed is not a concern. Alternatively, we create a pyramid of feature and use them for object detection (the right diagram). However, feature maps closer to the image layer composed of low-level structures that are not effective for accurate object detection.

Region-based Fully Convolutional Networks**, or** R-FCNs, are a type of region-based object detector. In contrast to previous region-based object detectors such as Fast/Faster R-CNN that apply a costly per-region subnetwork hundreds of times, R-FCN is fully convolutional with almost all computation shared on the entire image.

To achieve this, R-FCN utilises position-sensitive score maps to address a dilemma between translation-invariance in image classification and translation-variance in object detection. In contrast to previous region-based detectors such as Fast/Faster R-CNN that apply a costly per-region subnetwork hundreds of times, our region-based detector is fully convolutional with almost all computation shared on the entire image. To achieve this goal, we propose position-sensitive score maps to address a dilemma between translation-invariance in image classification and translation-variance in object detection. Our method can thus naturally adopt fully convolutional image classifier backbones, such as the latest Residual Networks (ResNets), for object detection.

2.1.3. Techniques for improving detectors

Various techniques for improving the performance of single-stage and two-stage detectors have been proposed in the past. The simplest is to clean the training data for faster convergence and for medium accuracy. The hard negative sampling technique is generally used to provide negative samples to achieve high final precision. Altering contextual information is another method used to improve detection accuracy or speed. to enrich the context of coarser features for better express object detection. BlitzNet improves SSD by adding semantic segmentation layer to achieve high detection accuracy . The object detection architectures discussed so far have several open source models pre-trained on large datasets, such as ImageNet [14], COCO [15] and ILSVRC [16]. These open source models have greatly benefited the field of computer vision and can solve specific object recognition problems with a few small extensions, thus avoiding any from scratch.These models differ in basic architecture, number of layers, inference speed, memory consumption, and detection accuracy. In order to enforce mask wearing in public areas to limit the spread of coronavirus in the community, machine learning methods based on available pretrained models are strongly recommended for the benefit of society. These pre-trained models -trained had to be refined 4,444 times using the benchmark dataset.

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. The project has been instrumental in advancing computer vision and deep learning research. The data is available for free to researchers for non-commercial use. The ImageNet dataset contains 14,197,122 annotated images according to the WordNet hierarchy. Since 2010 the dataset is used in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), a benchmark in image classification and object detection. The publicly released dataset contains a set of manually annotated training images. A set of test images is also released, with the manual

annotations withheld. ILSVRC annotations fall into one of two categories: (1) image-level annotation of a binary label for the presence or absence of an object class in the image, e.g., “there are cars in this image” but “there are no tigers,” and (2) object-level annotation of a tight bounding box and class label around an object instance in the image, e.g., “there is a screwdriver centered at position (20,25) with width of 50 pixels and height of 30 pixels”. The ImageNet project does not own the copyright of the images, therefore only thumbnails and URLs of images are provided.

- Total number of non-empty WordNet synsets: 21841
- Total number of images: 14197122
- Number of images with bounding box annotations: 1,034,908
- Number of synsets with SIFT features: 1000
- Number of images with SIFT features: 1.2 million

COCO is a large-scale object detection, segmentation, and captioning dataset. COCO is a large-scale object detection, segmentation, and captioning dataset. This version contains images, bounding boxes, labels, and captions from COCO 2014, split into the subsets defined by Karpathy and Li (2015). This effectively divides the original COCO 2014 validation data into new 5000-image validation and test sets, plus a "restval" set containing the remaining ~30k images. All splits have caption annotations. COCO Captions contains over one and a half million captions describing over 330,000 images. For the training and validation images, five independent human generated captions are be provided for each image.

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) evaluates algorithms for object detection and image classification at large scale. One high level motivation is to allow researchers to compare progress in detection across a wider variety of objects -- taking advantage of the quite expensive labeling effort. Another motivation is to measure the progress of computer vision for large scale image indexing for retrieval and annotation. Every year of the challenge there is a corresponding workshop at one of the premier computer vision conferences. The purpose of the workshop is to present the methods and results of the challenge. Challenge participants with the most successful and innovative entries are invited to present. Please visit the corresponding challenge page for workshop schedule and information. The rise in popularity and use of deep learning neural network techniques can be traced back to the innovations in the application of convolutional neural networks to image classification tasks.

Some of the most important innovations have sprung from submissions by academics and industry leaders to the *ImageNet Large Scale Visual Recognition Challenge*, or *ILSVRC*. The ILSVRC is an annual computer vision competition developed upon a subset of a publicly available computer vision dataset called ImageNet. As such, the tasks and even the challenge itself is often referred to as the ImageNet Competition.
The definition of top-down processing is the process of applying preexisting schemas and memories to new sensory information, in order to interpret this data more efficiently and accurately. This psychological construct is meant to help explain the relationship between sensation, the stimuli received by various receptors throughout the body, and perception, how this information is interpreted by the brain.

This theory argues that, when information about stimuli is received by the brain, a hypothesis is made about this new scenario using the context and past experience, without having to evaluate every feature of the information being received. In this way, the information flows from the top, down, hence the phrase top-down. These perceptions help interpret the new information.

British psychologist Richard Gregory introduced the concept of top-down processing in 1970. He argued that sensory information on its own is insufficient to create perception as humans experience it, because most of the information is lost by the time it arrives at the brain. Thus, the brain must depend on prior knowledge and experiences to properly process the information.

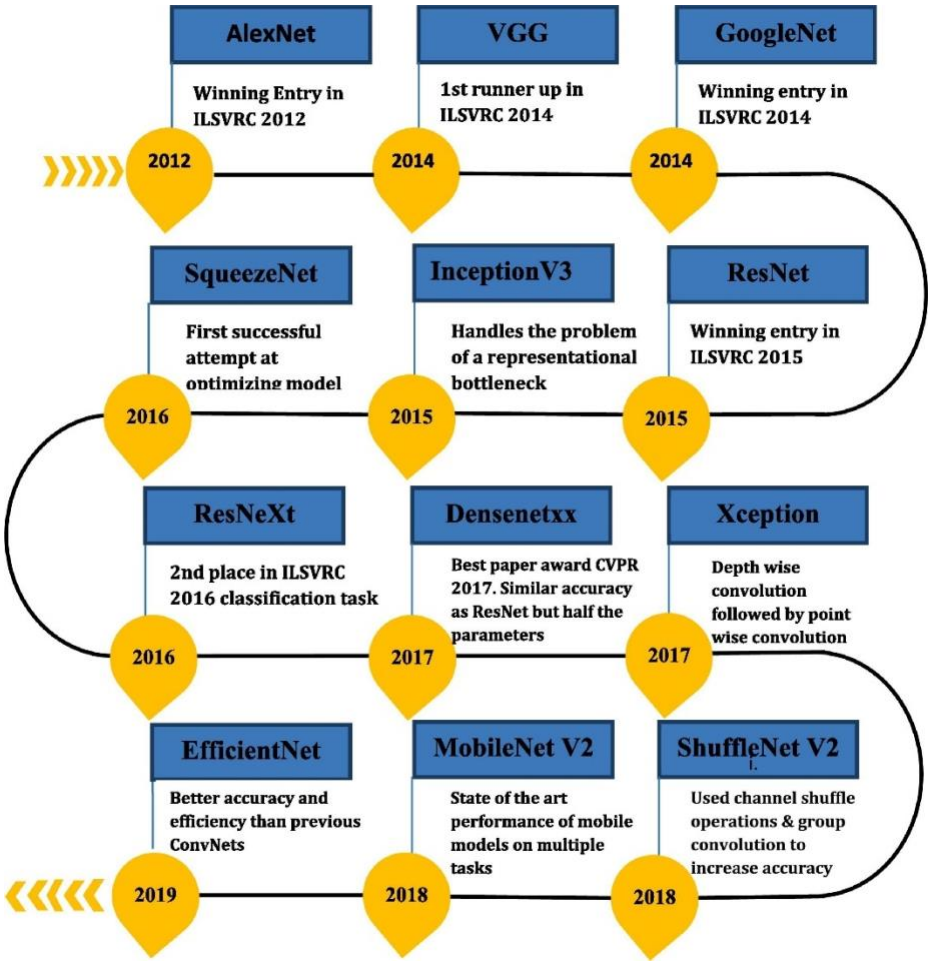


Fig 2.1. Various Pre-trained Models based on CNN Architectures.

To enforce mask over faces in public areas to curtail community spread of Coronavirus, a machine learning approach based on the available pre-trained model is highly recommended for the welfare of the society. These pre-trained models are required to be finely tuned with benchmark datasets. The number of datasets with diverse features pertaining to human faces with and without mask are given in Fig.2.2 .

Type of Datasets	Dataset	Scale	#Faces	#masked face images	Occlusion
Masked face detection Datasets	FDDB [31]	2845	5171	–	–
	MALF [32]	5250	11931	–	✓
	calebA [33]	200000	202599	–	–
	WIDERFACE [34]	32203	194000	–	✓
Face masked datasets	MAFA [35]	30811	37824	35806	✓
	RMFRD [36]	95000	9200	5000	✓
	SMFRD [36]	85000	5000	5000	✓
	MFDD [36]	500000	500000	24771	✓

Fig.2.2 Different Categories of Datasets.

The MALF dataset is a large dataset with 5,250 images annotated with multiple facial attributes and it is specifically constructed for fine grained evaluation.

Face Detection Data Set and Benchmark (FDDB), a data set of face regions designed for studying the problem of unconstrained face detection. This data set contains the annotations for 5171 faces in a set of 2845 images taken from the Faces in the Wild data set.

WIDER FACE dataset is a face detection benchmark dataset, of which images are selected from the publicly available WIDER dataset. We choose **32,203** images and label **393,703** faces with a high degree of variability in scale, pose and occlusion as depicted in the sample images. WIDER FACE dataset is organized based on 61 event classes. For each event class, we randomly select 40%/10%/50% data as training, validation and testing sets. We adopt the same evaluation metric employed in the PASCAL VOC dataset. Similar to MALF and Caltech datasets, we do not release bounding box ground truth for the test images. Users are required to submit final prediction files, which we shall proceed to evaluate.

A close look at the available datasets of faces reveals that there are mainly two types of datasets. These are: i) masks and ii) masked dataset. The masked face dataset is more focused on including images of faces with varying degrees of facial expressions and landmarks, while the mask-centric dataset includes face images primarily characterized by occlusions and their presence in the nose and position coordinates near the mouth area. After critically reviewing the available literature, the following gaps have been identified:

1. Although there are several open source models pre-trained on benchmark datasets, there are currently a few models capable of handling COVID-Related Face Mask Treatment Dataset.

2.The available mask datasets are sparse and require augmentation around different types of masks with different degrees of closure and semantics.

3. Although there are two main types of modern object detectors: single-stage detectors and two-stage detectors. But none of these really meet the demands of real-time video surveillance equipment.These devices are limited by less computing power and memory[37].Therefore, they need optimized object detection models that can perform real-time monitoring with less memory consumption and without significant loss of accuracy. The single-stage detector is suitable for real-time monitoring but is limited by low accuracy, while the two-stage detector can easily produce accurate results on complex inputs at the expense of computation time. All these factors necessitate the development of an integrated model for the monitoring device that can bring advantages in terms of computation time and accuracy.

To solve these problems, a deep learning model based on the Transfer learning trained on a highly tuned, CCTV compatible custom mask data set is proposed and discussed in detail in proposed system section.

2.2 Proposed System

The proposed model is based on the object recognition benchmark given in [38]. According to this criterion, all tasks related to object recognition problem can be summarized into three main parts: spine, neck and head. Here, the backbone corresponds to a basic convolutional neural network, which is able to extract information from images and convert it into feature maps. In the proposed architecture, the concept of transfer learning is applied to the backbone to extract new features for the model using the features learned from a powerful pre-trained convolutional neural network runs an exhaustive backbone building strategy using three popular pre-trained models, namely ResNet50, MobileNet, and AlexNet, to achieve the best results for face mask detection. ResNet50 proved to be an optimal choice to build the backbone of the proposed model. New to our work is the assembly of the neck. The central part, the neck, contains all the pre-processing tasks necessary before the actual classification of the image. To make our model compatible with monitors, Neck applies a different pipeline during the training and deployment stages. The training pipeline follows the creation of an unbiased custom dataset and the refinement of ResNet50. The deployment of the pipeline consists of extracting the live frames from the video, followed by the detection and extraction of the faces. To achieve a trade-off between the accuracy of face detection and the computation time, we propose an Image complexity predictor. The last component, Head, represents an identity detector or predictor, which can achieve the goals required for deep learning neural networks. In the proposed architecture, trained mask classifiers obtained after transfer learning are applied to detect masked and unmasked faces. The ultimate goal of law enforcement wearing masks in public places can only be achieved after restores personal face identification, which violates mask standards. Other actions may be taken depending on government/office policy. Since can have face size and orientation differences in slice ROIs, OpenFace 0.20 [17] is used to apply a affine transformation to recognize faces. A detailed description of each task in the proposed architecture is given in the Methodology subsection.

CHAPTER-3

Design flow/Process

3.1 Concept Generation

Following critical observations of the available systems in a literature review of existing systems, the following shortcomings were revealed:

1. Although there are several pretrained open source models on benchmark datasets, there are There are currently a handful of models capable of doing this. process mask datasets related to COVID-19.

2. The available mask datasets are sparse and require augmentation around different types of masks with varying degrees of closure and semantics.

3.Although there are two main types of modern object detectors: single-stage detectors and two-stage detectors. But none of the s really meet the demands of real-time video surveillance equipment. These devices are limited by less computing power and memory [37]. Therefore, they need optimized object detection models that can perform real-time monitoring with less memory consumption and without significant loss of accuracy. The single-stage detector is suitable for real-time monitoring but is limited by low accuracy, while the two-stage detector can easily produce accurate results on complex inputs at the expense of computation time. All these factors necessitate the development of an integrated model for the monitoring device that can bring advantages in terms of computation time and accuracy .

To solve these problems, a deep learning model based on the Transfer training trained on a highly tuned, CCTV-compatible custom mask data set is proposed and discussed in detail.

3.2 Objectives

The main objectives of the proposed works are as follows:

1. Develop a new object detection method combining single-stage and two-stage detectors to accurately locate objects from real-time video streams through the back-end transfer learning.

2. Development of an improved affine transformation to segment facial regions with face size, orientation, and background differences from unsupervised real-time images. This step made it possible to better locate the people who violated the standards for wearing a mask in public spaces/offices.

3. Create an unbiased mask dataset with an imbalance ratio of , which is almost one.

4. The proposed model requires less memory, which facilitates its deployment in embedded devices for monitoring purposes.

3.3. Proposed Architecture

The proposed model is based on the object recognition benchmark. According to this benchmark, all the tasks related to an object recognition problem can be ensembled under three main components: Backbone, Neck and Head as depicted in Fig. 2. Here, the backbone corresponds to a baseline convolutional neural network capable of extracting information from images and converting them to a feature map. In the proposed architecture, the concept of

transfer learning is applied on the backbone to utilize already learned attributes of a powerful pre-trained convolutional neural network in extracting new features for the model.

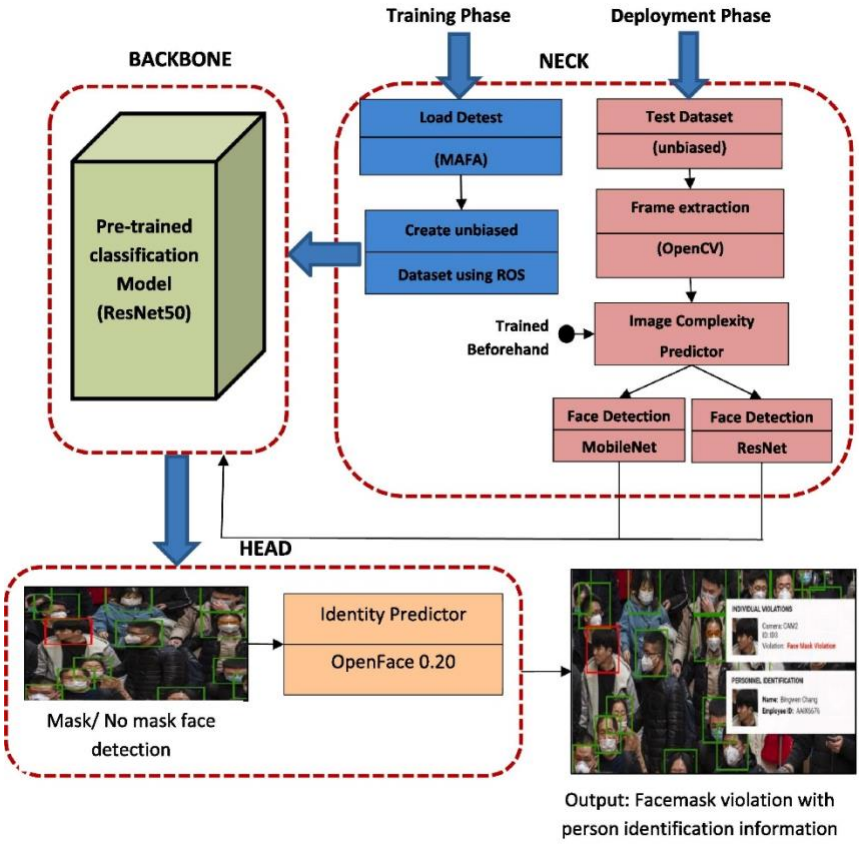


Fig 3.1 . Proposed Architecture.

An exhaustive backbone building strategy with three popular pre-trained models namely ResNet50, MobileNet and AlexNet are conducted for obtaining the best results for facemask detection. The ResNet50 is found to be optimized choice for building the backbone the proposed model. The novelty of our work is being proposed in the Neck component. The intermediate component, the Neck contains all those pre-processing tasks that are needed before the actual classification of images. To make our model compatible with surveillance devices, Neck applies different pipelines for the training and deployment phase. The training pipeline follows the creation of an unbiased customized dataset and fine-tuning of ResNet50.

The deployment pipeline consists of real-time frame extraction from video followed by face detection and extraction. In order to achieve trade-off between face detection accuracy and computational time, we propose an image complexity predictor. The last component, Head stands for identity detector or predictor that can achieve the desired objective of deep-learning neural network. In the proposed architecture, the trained facemask classifier obtained after transfer learning is applied to detect mask and no mask faces. The ultimate objective of enforcement of wearing of face mask in public area will only be achieved after retrieving the personal identification of faces, violating the mask norms. The action can further, be taken as per government/ office policy. Since there may exist differences in face size and orientation in cropped ROI, affine transformation is applied to identify facial using OpenFace 0.20 . The detailed description of each task in the proposed architecture is given in the following subsections.

The following methods for different subtasks are used to achieve the desired goal:-

3.3.1. Creating an unbiased mask dataset

MAFA (MAsked FAces) is a masked face detection benchmark dataset, of which images are collected from Internet images. MAFA contains 30,811 images and 35,806 masked faces. Faces in the dataset have various orientations and occlusion degrees, while at least one part of each face is occluded by mask. In the annotation process, each image contains at least one face occluded by various types of masks, while the six main attributes of each masked face, including locations of faces, eyes and masks, face orientation, occlusion degree, and mask type. The MAFA mask-centric dataset has a total of 25,876 images divided into two classes, masked and unmasked, initially thought to be. The number of hidden images in MAFA is 23,858 while the number of unmasked images is only 2018. It should be noted that

MAFA has an outer class imbalance issue, which may cause to be biased in favour of the majority class.Therefore, an ablation study is performed to analyze the performance of a image classifier, using the original (biased) MAFA set once, and then using the proposed (unbiased) data set. biased).

3.3.2. Supervised Pre-training

We discriminately pre-train CNNs on the biased original MAFA dataset.Pre-training is done using the open-source python library Caffe . In short, our CNN model almost matches the performance of the of Madhura et al. achieved a top-1 error rate of 1.8% on the MAFA validation set. This discrepancy may be due to the simplified training method of the Supervised Pretraining with Domain-Specific Fitting Another approach is to first remove the inherent bias present in the available datasets and then perform supervised training on the domain-specific balanced datasets. The bias is mitigated by applying random oversampling (ROS) and data augmentation.This technique reduces the imbalance ratio $\rho = 11.82$ (raw) to $\rho = 1.07$. The formula used to calculate the unbalance rate is given by equation:-

$$\rho = \text{Count}(\text{majority}(Di)) / \text{Count}(\text{minority}(Di))$$

Here D refers to the image dataset, and majority(Di) and minority(Di) return the classes majority and minority of D. Count(X) returns the number of images in any class x. After data balancing, the stochastic gradient descent (SGD) training with CNN parameters at a learning rate of 0.003 is set to on wrapped region proposals. The low learning rate makes it possible to improve the model without breaking the initialization. We added 2025 negative windows and 50 background windows to increase the unmasked dataset $\approx$ 22 KB. Balancing resulted in a 3.7% reduction in the error rate of the top 1. 5.3. Fine-tuning pre-trained models - In the presented work, face mask detection is implemented by deep neural networks due to their better performance than other classification algorithms. But training a deep neural network is expensive because it is a time-consuming task and requires high computing

power. In order to train the network faster and more economically, transfer learning based on deep learning is applied here. Transfer learning allows trained knowledge to be transferred from a neural network to a new model based on parameter weights. This improves the performance of new models, even when trained on small sets of data. There are several pre-trained models such as MobileNet, ResNet50, AlexNet etc. These are images formed with 14 million images from the ImageNet dataset . Of the models offered by, ResNet50 was chosen as the pre-training model for the classification of face masks. The latest ResNet50 layer is refined by adding five new layers. The newly added layers include a medium pooling layer with a pool size equal to 5×5, a smoothing layer, a dense ReLU layer consisting of 128 neurons, a dropout of 0.5, and a softmax with Enable Decision Layer of the function.

3.3.3. Image complexity predictor for face detection

To solve problem 3 identified in section 2, various face images are analyzed in terms of processing complexity. It should be noted that the dataset we are mainly considering, contains two main classes, masked and unmasked classes , but the masked class also contains various inherent occlusions in addition to surgical/tissue masks, such as ROI of that obstruct other objects, such as people, hands, hair, or certain foods. It turns out that these closures will affect the performance of face and mask detection. Therefore, finding the best compromise between accuracy and computation time for face detection is not trivial. Therefore, an image complexity predictor is proposed here. Its purpose is to divide the data into soft and hard images at an initial level and then perform masked and maskless classification at a higher level by a mask classifier.The important question we need to answer is how to determine if an image is soft or hard. The answer to this question is given by "Strategies for semi-supervised object classification" by Lonescu et al. The semi-supervised object classification strategy is suitable for our task because it predicts objects without locating them. To implement this strategy, we sample three sets of images: the first set (L) contains labeled training images (hard/soft), the second set (U) contains

unlabeled training images, and the third set (T) contains unlabeled training images. test pictures.We further apply the class learning method proposed in , which works iteratively by training hard/soft predictors on the expanded L training set at each iteration. The training set L is obtained by randomly moving k samples from U to L. We stop the training process when L reaches three times its original size . Initially fill L with 500 labeled samples. The initial labeling of the samples in L is done using the three most relevant image features that complicate the image.These attributes are the object densities (including full, truncated, and occluded planes), the average area covered by the objects normalized by the image size, and the resolution of the image. were evaluated by human annotators. We employ 50 reliable annotators, each displaying 10 images.

Fig. 3.2. Variety of Occlusions Present in Dataset.

We asked two questions to each annotator. The question is, "Are there people in the picture?"and "How many faces are there in the given images, including full, cropped, and closed faces?" ". We ensure that the annotation task is non-trivial by presenting images in random order, so that if the answer is yes for one image, it may be no for another image. We save each annotation times the respondent answered the question We removed all answers longer than 30 seconds times to avoid bias.Additionally, the response time of each annotator was normalized by subtracting it from the interval time and dividing it by standard deviations. We calculated the geometric mean of all responses times per frame and stored these values as object densities. We further observe that the complexity of the image is positively correlated with the density of the object and negatively correlated with the object size and image resolution. Based on these image features, each image is assigned a Ground Truth Visibility Difficulty Score. Vector Regression as described in .The last layers of VCG-f are replaced by fully connected layers. Each test image is divided into three bins of sizes 1x1, 2x2, and 3x3 to achieve a pyramid representation of the image for better performance. The image is also flipped horizontally by and the same pyramid is placed on it. The 4096, features extracted from each bin are combined to obtain a single vector of features, which is then normalized using the L2 norm. The resulting normalized feature vectors are further used to regress the complexity score of the image.Therefore, the model automatically predicts the complexity of the image for each T-frame. After identifying the hardness of the test images using an image complexity predictor, soft images are proposed for processing by a single-stage fast detector, while hard images are accurately processed by a two-stage detector. We use the MobileNet-SSD model to predict the class for soft images, and a faster CNN R- based on ResNet50 for hard images. The algorithm of the Image Complexity Predictor is detailed as follows:

Algorithm:-

Image_Complexity_predictor()

1.Input:

2. Image ← Input Image

3. D_{fast} ← Single Phase Detector

4. D_{slow} ← Two Phase Detector

5. C ← Image Complexity

6.Calculation:

7. If(C=Soft) R ← D_{slow}(Image)

8. else R ← D_{fast}(Image)

9. Output:

10. R ← 4 combinations of region4.

By dividing the test dataset into different parts of pro- images processed by each detector, from pure Mobi- leNet (100-0%) to three intermediate parts (75-25%, 50-50%, 25-75%)) in pure ResNet50 (0–100%).Here, the test data is based on randomization or soft and hard waste splits, given by the image complexity predictor . To reduce bias, the average map over 5 runs is a random overflow of records. Elapsed time is measured on an IntelI7, 2.5 GHz processor and 16 GB of RAM.

Comparison Parameters	**MobileNet-SSD to ResNet50 (Left to Right)**			
	100–0%	**75–25%**	**50–50%**	**25–75%**
Random split (mAP)	0.8868	0.9095	0.9331	0.9650
Soft/hard split (mAP)	0.8868	0.9224	0.9631	0.9892
Image complexity prediction time (ms)	-	0.05	0.05	0.05
Mask detection time (ms)	0.05	1.92	3.08	5.07
Total Computation Time (ms)	0.05	1.97	3.13	5.12

Table 3.1. Summarizes mAP score and Computation time for various combinations of MobileNet and ResNet50 over test dataset.

3.3.4. Identity prediction

After detecting faces with masks and non-mask in the search proposal, the non-mask faces are passed separately into a neural network for further exploration of a person's identity for being violating the facemask norm. The step requires a fixed-sized input. One possible way of getting a fixed-size input is to reshape the face in the bounding box to 96 × 96 pixels. The potential issue with this solution is that the face could be looking in a different direction. Affine transformation can handle this issue very easily. The technique is similar to deformable part models described in. The use of affine transformation is depicted in 4.

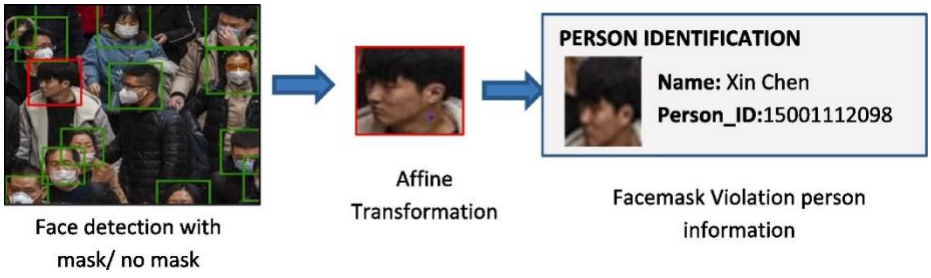


Fig. 3.3. Affine Transformation for localizing the face with no mask.

CHAPTER-4

RESULT ANALYSIS AND VALIDATION

4.1. EXPERIMENTAL SETUP

Experiments were set up by loading different pre-trained models using the Torch Vision package (https://github.com/pytorch/vision). These models were enhanced on our dataset using the open source Caffe Python library. We choose our unbiased custom dataset of 45,000, images, available online at https://www.kaggle.com/mrviswamitrakaushik/facedatahybrid. The Int-Scenario training strategy is adopted by as used in [8]. The data set is divided into training set, test set, and validation sets, which are 64:20:16 respectively. The algorithm is implemented with Python 3.7, and face detection is implemented by MobileNet-SSD/ResNet.dib mask for learning detection rate=0.003, momentum=0.9 and lot size=64.

4.2. Model comparison

We can apply transfer learning on pre-trained models for image classification but one question that yet to answer is how we can decide which model is effective for our task. In this section, we will compare three efficient models viz. ResNet50, AlexNet and MobileNet, based on following criteria:

- 1. Top-1 Error: This type of error occurs when the class predicted with the highest confidence is not the same as the true class.

- 2. Inference Time on CPU: It is the time taken by the model to predict the class of input image, that is starting from reading the image, performing all intermediate transformations and finally generating the high confidence class to which the image belongs.
- 3. Number of Parameters: It is the total count of learnable elements present in all the layers of a model. These parameters directly contribute to prediction capability, model complexity and memory usage. This information is very useful for understanding the minimum amount of memory required for each model. Further, it had been analysed by Simone Bianco et. al. that we require optimum number of learnable parameters so that trade-off between model accuracy and memory consumption may be achieved.

A model with minimum Top-1 error, less inference time on CPU and optimum number of parameters will be considered as a good model for our work.

The confusion matrices for different models during testing are given in 5. The accuracy comparison of various models based on Top-1 error is presented graphically in 6(a). It may be noted from the graph that the error rate is high in AlexNet and least in ResNet50. Next, we compared the model based on inference time. Test images are supplied to each model and inference times for all iterations are averaged out. It may be observed from 6(b) that MobileNet takes more time to infer images whereas ResNet and AlexNet take almost equal inference time for images. Further, the memory usage comparison among underlying models is done by finding the number of learnable parameters. These parameters can be obtained by generating model summary in Google colab for each model. It may be noted

in 6(c) that the number of parameters present in AlexNet is around 27.55 million for our customised dataset. Furthermore, the number of parameters present in MobileNet and ResNet 50 are around 3.4 million and 25.5 million respectively.

Fig. 4.1. Confusion Matrix Obtained for Various Pre-trained Models.

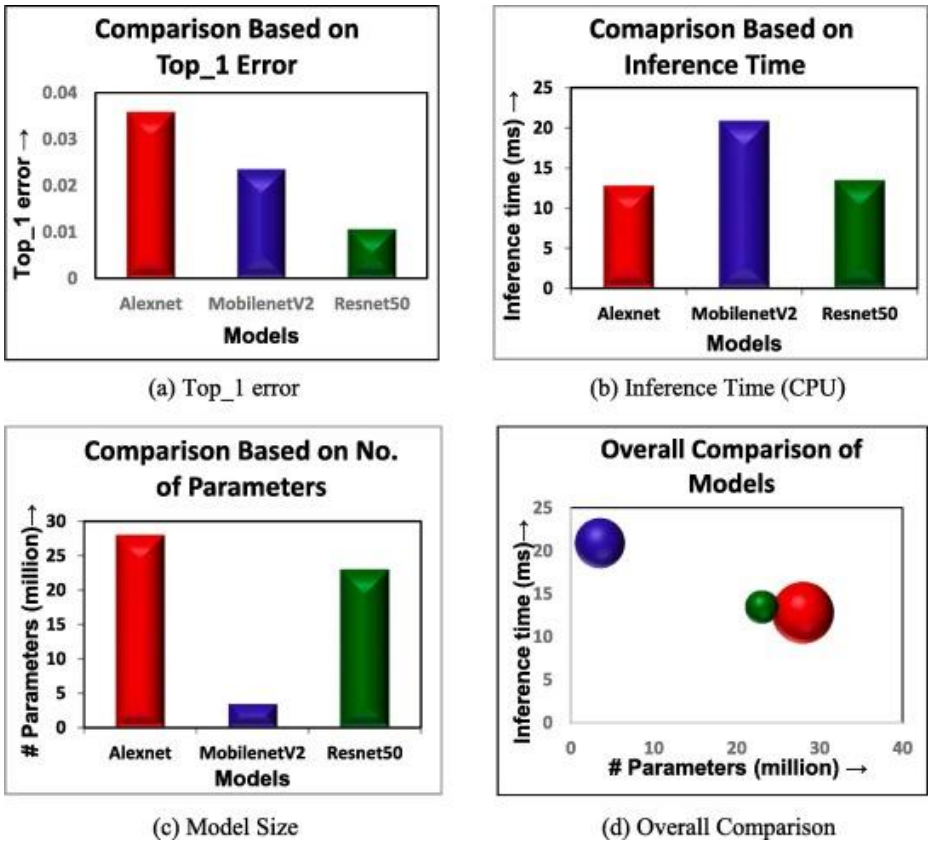


Fig. 4.2. Comparison of Various Models on Different Performance Criteria.

It may be observed from Fig. 7, smaller bubbles are better in terms of accuracy and bubbles near the origin are better in terms of memory usage and inference speed. Now, the answer to **RQ1** can be given as follows:

- AlexNet has a high error rate.
- MobileNet is slow in inferring results.
- ResNet50 is an optimized choice in terms of accuracy, speed and memory usage for detecting face mask using transfer learning.

4.3. Performance analysis of image complexity predictor

For performance evaluation of the Image complexity predictor, we use Kendall's coefficient τ (tau). We compute Kendall's rank correlation coefficient τ between the predicted image complexity score and ground truth visual difficulty score. The Kendall's rank correlation coefficient is a suitable measure for our analysis because it is invariant to different ranges of scoring methods. Based on image properties, each human annotator assigns a visual difficulty score to an image from a range that is different from the range, predicted image complexity score is assigned. The Kendal's rank correlation coefficient is computed in Python using kendalltau()SciPy function. The function takes two scores as arguments and returns the correlation coefficient. Our predictor attains Kendall's rank correlation coefficient τ of 0.741, implying the remarkable performance of the image complexity predictor. It may be observed from Fig. 7 that a very strong correlation exists between ground truth and predicted complexity scores.

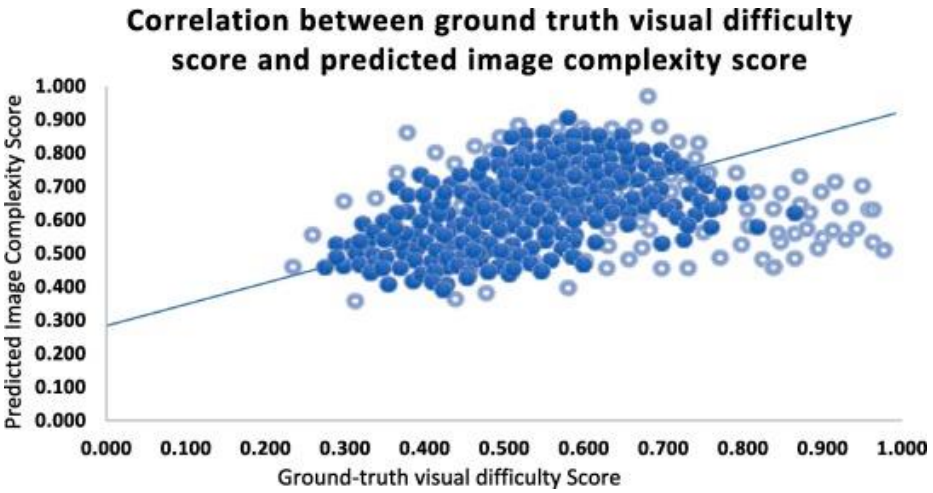


Fig. 4.3. Correlation between Ground Truth Visual Difficulty Score and Predicted Image Complexity Score.

4.4. Performance analysis of identity predictor

In order to impose wearing of face mask in public areas such as schools, airports, markets etc., it becomes essential to find out the identity of those faces which are violating the rules, means either not wearing or not correctly wearing a face mask. Typically, these identities can be found by training our model with persons faces. For this purpose, the photographs of 2160 students are collected and populated in our customized dataset which is available online at https://www.kaggle.com/facemaskdeeplear/facedatahybrid. In order to well-train our system, we have taken five photographs of each student, ensuring face looking in different directions with different backgrounds. To further, proceed with the experiment, the video streaming from four CCTV cameras located at different

locations in Department of Computer Applications, Chandigarh University, Punjab, India is analysed. We captured the images from real-time video.

Precision and Recall are taken as evaluation metrics for identity prediction. The Precision and Recall for identity predictor are 98.86% and 98.22% respectively.

4.5. Comparison of proposed model with existing models

In this section, we aim to compare the performance of the proposed model with public baseline results published in RetinaFaceMask[11], which aims to answer RQ2. Since RetinaFaceMask is trained on the MAFA dataset and performance is evaluated using precision and recall for face and mask detection so, for comparison purposes, the performance of the proposed technique is also evaluated in the same environment. We employed two standard metrics namely Precision and Recall for comparing the performance of these two systems. The experimental results are reported in Table 4.1. It may be noted from Table 4.1 that the proposed model with ResNet50 as backbone achieves higher accuracy as compared to RetinaFaceMask.

Model	Face Detection		Mask Detection	
	Precision (%)	Recall (%)	Precision (%)	Recall (%)
RetinaFaceMask based on MobileNet	83.0	95.6	82.3	89.1
RetinaFaceMask based on ResNet	91.9	96.3	93.4	94.5
Proposed model based on ResNet50	99.2	99.0	98.92	98.24

Table 4.1. Comparison of Proposed model with Recent face mask detection Model

Particularly, the proposed model generates 11.75% and 11.07% higher precision in the face and mask detection respectively when compared with RetinaFaceMask. The recall is improved by 3.05% and 6.44% in the face and mask detection respectively.

CHAPTER-5

CONCLUSION

In this work, a deep learning-based approach is proposed to detect above-face masks in public places to limit the community spread of the Corona- virus. The proposed technique efficiently handles occlusions in dense cases using a set of one- and two-stage -class detectors at the preprocessing level. The ensemble method not only helps the achieve high accuracy, but also greatly improves the detection speed of the . In addition, applying transfer learning to the model trained before and carrying out experiments Extensive investigations on the unbiased data set result in a very robust and inexpensive system. Face ID detection that further violates the mask specification increases the public welfare utility of the system. In the end, this work opens interesting future perspectives for researchers. Firstly, the proposed technique can be integrated into any high resolution video surveillance equipment and is not limited to mask detection. Second, the model can be extended to detect landmarks on faces with masks for biometric purposes. The identity detection of faces, violating the mask norms further, increases the utility of the system for public benefits. To conclude:-

- In this work, a deep learning based model for detecting masks over faces in public place to curtail community spread of Coronavirus is presented. The proposed model efficiently handles varying kinds of occlusions in dense situation by making use of ensemble of single and two stage detectors. The ensemble approach not only helps in achieving high accuracy but also improves detection speed considerably. The model is 98.2% accurate at mask detection with average inference times of 0.05 seconds per image.

- The high accuracy of model is also due to highly balanced face mask centric dataset achieved through Random over-sampling with data augmentation over original MAFA dataset. Our technique reduces the imbalance ratio ρ = 11.82 (original) to ρ = 1.07.

- The other factors that contributed towards achievement of highly efficient model include application of bounding box affine transformation and transfer learning. The bounding box transformation improves localization performance during mask detection. Transfer learning leads to good results by enabling use of powerful pre-trained model such as ResNet 50 being trained on large dataset like ImageNet.

- The experiment is conducted with three most popular baseline models viz. ResNet50, AlexNet and MobileNet and explored the

possibility of plug-in with proposed model to achieve highly accurate results in less inference time. It is observed that proposed technique achieves high accuracy (98.2%) when implemented with ResNet 50.

- Besides, the results are also compared with recent public baseline model published as RetinaFaceMask [14] and improvements of 11.07% and 6.44% in Precision and Recall for mask detection are recorded.

Finally, the work opens interesting future directions for researchers. Firstly, the proposed technique can be integrated into any high-resolution video surveillance devices and not limited to mask detection only. Secondly, the model can be extended to detect facial landmarks with a facemask for biometric purposes.

REFERENCES

1. https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200816-covid-19-sitrep-209.pdf?sfvrsn=5dde1ca2_2

2. PAHO/WHO — Pan American Health Organization. (n.d.). https://www.paho.org/en/news/2-6-2020-social-distancing-surveillance-and-stronger-health-systems-keys-controlling-covid-19.

3. L. Nanni, S. Ghidoni, S. Brahnam
   Handcrafted vs. non-handcrafted features for computer vision, 10.1016/j.patcog.2017.05.025

4. Y. Jia et al., Caffe: Convolutional architecture for fast feature embedding, in: MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia, 2014, doi: 10.1145/2647868.2654889.

5. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun, OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks, 2014.

6. D. Erhan, C. Szegedy, A. Toshev, D. Anguelov
   Scalable Object Detection using Deep Neural Networks
   Proceedings of the IEEE conference on computer vision and pattern recognition (2014), pp. 2147-2154, 10.1109/CVPR.2014.276

7. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, vol. 2016-Decem, pp. 779–788, doi: 10.1109/CVPR.2016.91.

8. M. Jiang, X. Fan, and H. Yan, RetinaMask: A Face Mask detector, 2020, http://arxiv.org/abs/2005.03950.
9. P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun, OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks, 2014.
10. Proceedings of the IEEE conference on computer vision and pattern recognition (2014), pp. 2147-2154, 10.1109/CVPR.2014.276
11. M. Jiang, X. Fan, and H. Yan, RetinaMask: A Face Mask detector, 2020, http://arxiv.org/abs/2005.03950.
12. R. Girshick, J. Donahue, T. Darrell, J. Malik
Region-based Convolutional Networks for Accurate Object Detection and Segmentation
IEEE Trans. Pattern Anal. Mach. Intell., 38 (1) (2015), pp. 142-158, 10.1109/TPAMI.2015.2437384
13. K. He, X. Zhang, S. Ren, J. Sun
Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition
IEEE Trans. Pattern Anal. Mach. Intell. (2015), 10.1109/TPAMI.2015.2389824
14. J. Deng, W. Dong, R. Socher, L.-J. Li, Kai Li, Li Fei-Fei, ImageNet: A large-scale hierarchical image database, 2010, doi: 10.1109/cvpr.2009.5206848.
15. T. Y. Lin et al., Microsoft COCO: Common objects in context, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial

Intelligence and Lecture Notes in Bioinformatics), 2014, vol. 8693 LNCS, no. PART 5, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.

16. I.D. Apostolopoulos, T.A. Mpesiana

Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks

Phys. Eng. Sci. Med. (2020), 10.1007/s13246-020-00865-4

17. K. Chen et al., MMDetection: Open MMLab Detection Toolbox and Benchmark, 2019, arXiv preprint arXiv:1906.07155

(2019). http://bamos.github.io/2016/01/19/openface-0.2.0/