Multimedia Tools and Applications (2023) 82:34321–34361
https://doi.org/10.1007/s11042-023-14686-6

# Deep learning for face mask detection: a survey

Aanchal Sharma [1] · Rahul Gautam [1] · Jaspal Singh [1]

Received: 7 December 2021 / Revised: 16 June 2022 / Accepted: 3 February 2023 /
Published online: 4 March 2023


**Abstract**
The Coronavirus Disease (Covid-19) was declared as a pandemic by WHO (World Health Organization) on 11 March 2020, and it is still currently going on, thereby impacting tremendously the whole world. As of September 2021, more than 220 million cases and 4.56 million deaths have been confirmed, which is a vast number and a significant threat to humanity. Although, As of 6 September 2021, a total of 5,352,927,296 vaccine doses have been administered, still many people worldwide are not fully vaccinated yet. As stated by WHO, "Masks" should be used as one of the measures to restrain the transmission of this virus. So, to reduce the infection, one has to cover their face, and to detect whether a person's face is covered with a mask or not, a "Face mask detection system" is needed. Face Mask Detection falls under the category of "Object Detection," which is one of the sub-domains of Computer Vision and Image Processing. Object Detection consists of both "Image Classification" and "Image Localization". Deep learning is a subset of Machine learning which, in turn, is a subset of Artificial intelligence that is widely being used to detect face masks; even some people are using hybrid approaches to make the most use of it and to build an efficient "Face mask detection system". In this paper, the main aim is to review all the research that has been done till now on this topic, various datasets and Techniques used, and their performances followed by limitations and improvements. As a result, the purpose of this study is to give a broader perspective to a researcher to identify patterns and trends in Face mask detection (Object Detection) within the framework of covid-19.

**Keywords** COVID-19 · Face mask detection · Deep learning · Machine learning · Object detection · Convolutional neural network

✉ Rahul Gautam
rahulgautam@sliet.ac.in

[1] Department of Computer Science & Engineering, Sant Longowal Institute of Engineering & Technology, Longowal, Punjab, India

## 1 Introduction

The severe acute respiratory syndrome Coronavirus - 2 (SARS-CoV-2), currently known as "**COVID-19**", is genetically related to one of the Coronaviruses responsible for the SARS outbreak of 2003. [92] It is a member of the large coronaviruses (CoVs) family, which consists of various viruses such as 229E, NL63, OC43, HKU1, MERS-CoV (2012), and the original SARS-CoV(2003). Covid-19 has enormously impacted human lives, and it was declared a pandemic by WHO on 11 March 2020. As of 7 September 2021, more than 221 million cases and 4.57 million deaths have been confirmed. Initially, the cases of COVID-19 were identified in Wuhan City, China, in December 2019 [17, 65]. The virus is transmitted from one person to another when an infected person coughs, sneezes, speaks, or breathes. Humans are also infected by touching surfaces contaminated by the virus when they touch their eyes, nose, or mouth without first washing their hands [93]. Covid-19 is a novel virus with which we are not familiar in the past. Even though vaccines are now invented, and as of **6 September 2021**, a total of 5,352,927,296 vaccine doses have been administered, still "**Breakthrough infections**" (An infection of a fully vaccinated person) are expected. Therefore, vaccines are not 100% effective at preventing infection; some people who are fully vaccinated will still get COVID-19, So one must continue wearing a **face mask** and taking all the necessary precautions [84].

Due to this Pandemic, People are undoubtedly facing lots of problems, mainly from Physical health to mental health issues, Food hardships, Education, people losing their jobs, and impact on the global economy. Maybe the Pandemic will end soon, but the impact will surely last. As of 8 September 2021, more than 222 million cases and 4.59 million deaths have been confirmed, and this shows how Covid-19 negatively affected the whole world to a very great extent.

Figure 1 precisely shows how Coronavirus infected people all over the world. The blue line represents the cumulative total deaths, whereas the red line shows cumulative total corona cases. Different countries in the world are suffering from this infectious virus whereas the United States of America solely reported 45,635,708 cases (Nov 1, 2021) which is the highest

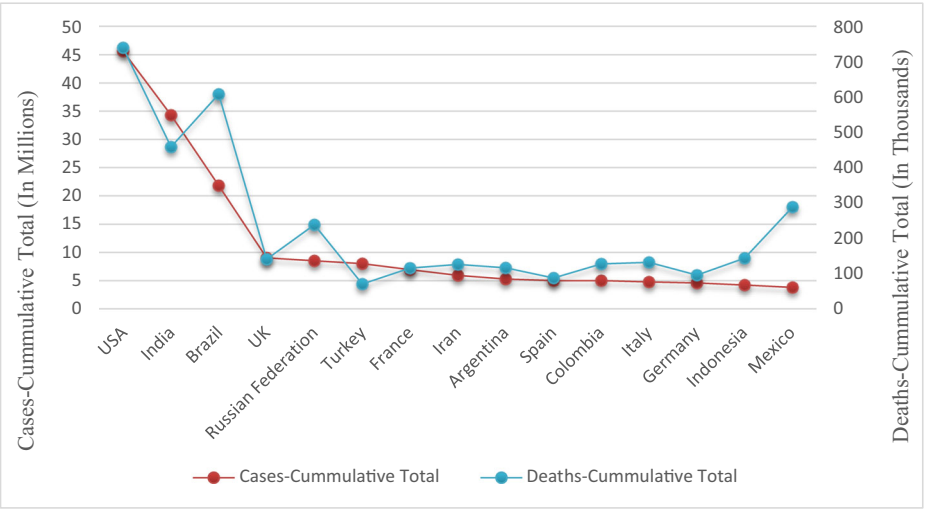



**Fig. 1** Total Corona Cases and Total Deaths were confirmed from different countries in the world in the time period of Jan 3, 2020, to Nov 1, 2021

among all the other countries as well death cases cover 47% of the total portion. Other countries such as India, Brazil, and The United Kingdom are some of the top infected countries facing difficulties due to Coronavirus.

In Fig. 2, total death Cases from the different regions across the world are represented in the form of the Doughnut Chart, in which America and Europe suffered more losses of lives, covering 47% and 28% of the total portion, respectively.

Day by day, COVID-19 cases are accelerating tremendously because people are not seen to follow the mandatory Covid norms such as physical distancing, wearing a mask, keeping rooms well ventilated, avoiding crowds, Sanitizing their hands, and coughing into a bent elbow or tissue [4]. Even though vaccines have now been developed but no vaccine is 100% effective, it just helps to boost the immunity. However, there is still a possibility of break-through infections, so according to WHO, even after being vaccinated, wearing a mask is necessary along with all other mandatory precautions to keep you as well as others safe—the WHO advises wearing a mask to reduce the spread of respiratory droplets containing infectious viral particles. N95 respirators, Surgical Masks, or Procedural Face Masks are some types of masks that help to prevent an infected person from transmitting the virus to others or prevent a healthy wearer from the infection [57]. Also, mask-wearing reduces the likelihood of other respiratory diseases, such as tuberculosis and influenza, occurring during the pandemic, which would then complicate or worsen the situation [45]. Various Awareness campaigns on facemasks are being held, and public places such as shopping malls, cinema halls, etc., are encouraging **"NO MASK, NO ENTRY".** In this Covid Situation, it is manually not possible to monitor each and everyone in large organizations and in crowded places to check whether a person is with or without a mask; here, the "Face Mask Detection System" is a lifesaver for us. Covid Pandemic is a recent area of interest for all the researchers, and various researches are being carried out for the same to keep the world safe.

Today, the field of Computer Vision is developing at a rapid pace. **Computer vision** is a branch of AI that allows computers to extract useful information from digital images and videos [16]. Computer vision consists of various sub domains such as Object Recognition,

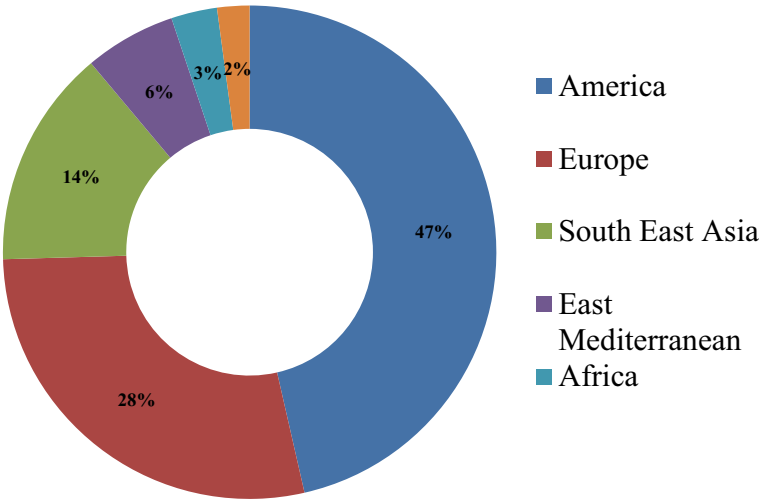



**Fig. 2** Doughnut chart depicts the total number of deaths reported by the World Health Organization from various regions worldwide. The reported number of deaths spans from January 27, 2020, to September 6, 2021

Object Segmentation, and Object Detection [28], the "Face Mask Detection Techniques" fall under the domain called "**Object Detection".** The ability of a computer to locate and identify objects in an image is referred to as object detection. Object Detection consists of two steps: **Object Localization** and **Image Classification**. In the Face mask detection Algorithm, the Object Localization task tries to identify the location of the face mask with the help of bounding boxes and then performs the image Classification task by classifying it into one of the categories say, "With-Mask" or "Without-Mask".

Earlier Object detection has been done using traditional Non- neural approaches which has certain shortcomings such as: In traditional ML detectors, Feature extractors were mainly hand-crafted which implies that Feature extraction has been done by a domain expertise that results in Low level feature extraction with Increase in time consumption, In these traditional algorithms, multi- sliding window is used which slides over the whole image that may results in redundant region proposals generation hence, it makes the process very complex and also these windows were designed manually and fixed in nature, but after the development of DCNN (deep convolutional neural networks) which has a deep structure with different layers (convolutional layers, pooling layers), Object detection with deep learning-based algorithms outperformed traditional algorithms due to its: Automatic low to highly complex feature extraction capability, Improved Accuracy, Increase in speed and DL-based algorithms can perform very well on large training dataset, Data augmentation technique is generally used to artificially increase the small dataset to achieve the better accuracy.

A typical Convolutional Neural Network (briefly discussed in Section 2) serves as the foundation for deep learning algorithms, and due to this, there are great improvements in performance. And nowadays, for face mask detection algorithms, there are some CNN architectures that are popularly used as network backbone such as AlexNet [44], VGGNet [78], GoogLeNet [81], Inception series [41, 82, 83], ResNet [35], DenseNet [40] and MobileNet [37]etc.

### 1.1 Contributions

As detection of masks came across into researcher's attention recently due to Covid-19, so there are very few surveys that have been published on face mask detection. Each of these surveys has its limitations, such as the lack of detailed information about face mask detection algorithms and information presented in these review papers are also not organized well. Most of the review papers are focused explicitly on recent deep learning-based algorithms, and they are not concentrated on the evolution of these algorithms from the traditional algorithms.

Even though extensive research work has been published on facemask detection approaches, there exists only a few review articles on face mask detection, such as **"A Review on Face Mask Detection using Convolutional Neural Network"** [3] and **"Face mask detection in COVID-19: a strategic review"** [87] but these contains a significant amount of shortcomings and an efficient and thorough review is still missing.

As in [3], firstly, they only reviewed a few literature studies that are primarily based on CNN-based algorithms. Secondly, the authors have entirely ignored the importance of the datasets USED in the research works and have not discussed them in the review paper. Apart from this, the review focuses on the discussion solely based on algorithms used for face mask detection and does not provide performance metric-based analysis for the considered studies. In another review [87], first, they have mainly discussed only about deep learning techniques. The authors did not provide the necessary background for traditional object detection

algorithms and their working were not discussed. In contrast, we present the fundamental framework for Non-Neural object detection algorithms with a step-by-step explanation, including deep learning techniques (neural network object detection algorithms) in a chronological order starting from 1999 to 2020. Second, only few models such as Faster RCNN, R-FCN, YOLO are discussed. Third, they focused only on how a dataset can be accessed and they have mentioned only four datasets, whereas, in our survey, we have provided 15 publicly available benchmark face mask detection datasets with a detailed description (Tables 1 and 2).

In comparison to these previously published review articles, the organization of this review is more apparent, and the material of each section is more clearly elaborated.

Given the recent development and research trends, a comprehensive and detailed analysis of existing face mask detection approaches to contribute more progress in the face mask detection techniques is the focus of this study. Our goal is to provide well organized and essential conceptual knowledge of core traditional object detection techniques as a basis for face mask detection and to define taxonomies of object detection approaches. Apart from this, a review of publicly available face mask detection datasets and suitable performance evaluation measures are also provided (Fig. 3).

Initially, to identify different papers, we used Google Scholar, Web of Science, Semantic Scholar, and CiteSeer to search for the term "face mask detection" and "Object detection" and "Deep learning for face mask detection". Although there are different review articles and research papers that have been available related to object detection which solely review about object detection systems that have been used for decades, there are limited review articles available on the face mask detection.

The main contributions of this paper are mentioned below:

- To present the essential background for face mask detection approaches using traditional non-neural and deep learning-based object detection methods.
- The survey presented state of the art face mask detection literature in chronological order where deep learning-based algorithms are categorized into two groups namely regional proposal based (two-stage) object detector and classification/regression based (one-stage) object detectors.
- The survey discusses mostly cited publicly available benchmark face mask detection datasets; researchers can easily choose suitable dataset from these datasets mentioned in this survey.
- All the existing work's results by using different evaluation criteria are presented systematically in this paper so that it provides an insight to a researcher to understand it better and make improvements in those results.

**Table 1** Comparative Analysis with existing Review papers on Face Mask Detection

| Review articles | Traditional background | Deep Learning-based Techniques | # Datasets reviewed | Experimental analysis for face mask detection | Representation of object detection algorithms |
|---|---|---|---|---|---|
| K. Adithya et al. [3] | No | Yes | – | No | – |
| Vibhuti et al. [87] | No | Yes | 4 | Yes | Hierarchical |
| Proposed Review | Yes | Yes | 15 | Yes | Chronological and Hierarchical |

**Table 2** Experimental Analyses of existing different Face mask algorithms are described in Section 3

| Ref. | Year | Evaluation Criteria | Dataset Used | Results |
|---|---|---|---|---|
| [61] | 2015 | Recall, False Positive Rate, ROC Curves | LFW(Labeled Faces in the Wild) | FPR was below 5%, and Recall was above 95%. |
| [9] | 2017 | IoU, Recall, Accuracy | MASKED FACE dataset | If IoU>0.5 i.e. Correct Prediction, Recall=87.8%,Accuracy=86.6% |
| [21] | 2019 | Accuracy | ORL face dataset | The average Accuracy of masked face image recognition=72%, and non-masked face recognition accuracy is on average 95%. |
| [74] | 2020 | mean average precision (mAP) | Moxa3K Benchmark Dataset | YOLOv3 832×832:mAP@50=61.73%, YOLOv3Tiny414×414:mAP@50=56.27%, SSD 300 MobliNetv2: mAP@50=46.52%, F-RCNN 300 Inceptionv2: mAP@50=60.5%. |
| [42] | 2020 | Accuracy, Precision, Specificity,(IoU) | Simulated Masked Face Dataset (SMFD) | During the Testing phase, Accuracy, Precision, Specificity, and IoU=100%. |
| [24] | 2020 | Precision, Recall | Face Mask Dataset(FMD) | RetinaFaceMask with MobileNet: For FACE Detection, precision and recall=83.0% and 95.6% respectively and for Mask detection, precision and recall=82.3% and 89.1% respectively. |
| [53] | 2021 | Average Precision | Medical Masks Dataset (MMD) and Larxel's Face Mask Detection Dataset (FMDD) | Average Precision=81% |
| [99] | 2021 | mean average precision (mAP) | MAFA:A Dataset of Masked Faces | Context-Attention R-CNN: mAP=84.1% |
| [54] | 2021 | Recall,Precision,F1 Score, Testing Accuracy | RMFD,SMFD, and LFW | Testing Accuracy=99.64%, 99.49%, 100% for RMFD, SMFD, and LFW datasets. |
| [38] | 2021 | Precision,Recall,F1Score | MaskedFace-Net,Flickr-Faces-HQ Dataset(FFHQ) and CelebA dataset | Detection Rate=98% |
| [91] | 2021 | Average Precision | WIDER FACE dataset and MAFA | Average Precision =89% |
| [59] | 2021 | Accuracy,F1 Score | Real-Time-Medical-Mask-Detection Dataset | Accuracy=92.64%,F1 Score=0.93 |

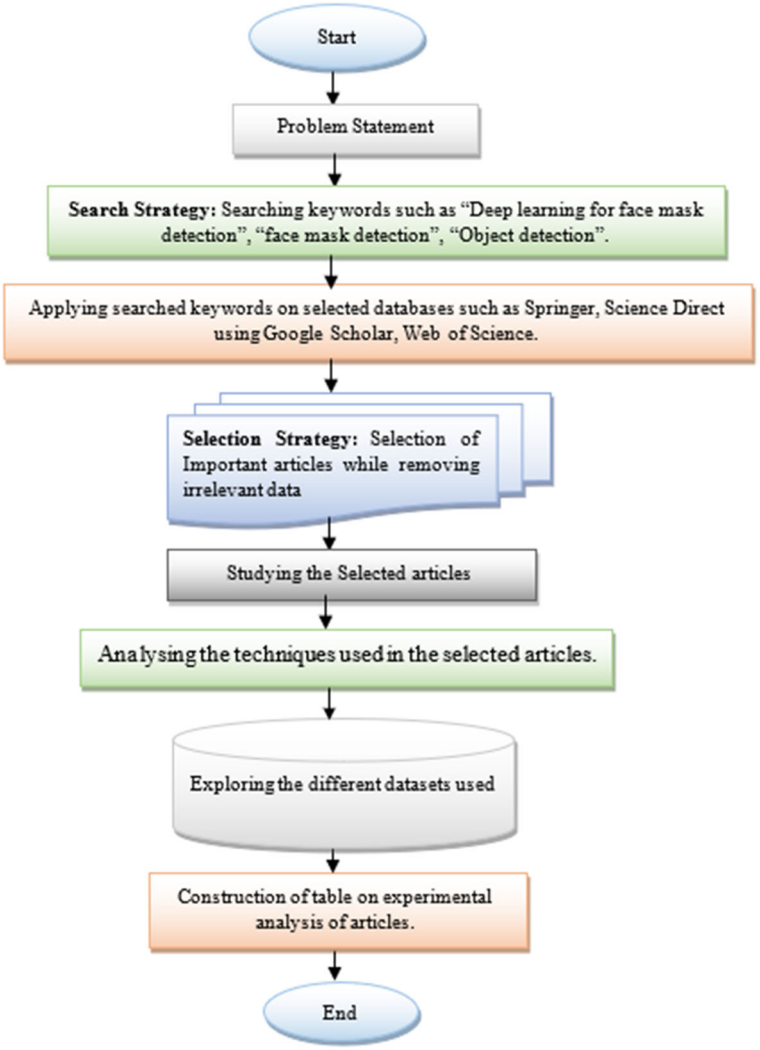



**Fig. 3** Flowchart of used review methodology

- In the end, the various application areas where the face mask detection system can be used are summarized, followed by various challenges faced in this area that need to be highlighted for future developments.

The rest of the review paper is systematically organized in the following sections. Section 2 provides a brief introduction to Object detection followed by different Object detection Methods, which include all the Non-Neural and Neural network approaches. Section 3 presents the existing methods adopted by the different authors for face mask detection before and after Covid-19. Section 4 outlines the popular datasets USED in the existing literature mentioned in Section 3. Section 5 reports the results obtained during the experimental evaluation of different face mask detection algorithms used in the previous Section 3. Section 6 highlights the various application areas of face mask detection systems, followed

by challenges faced and future scope in Section 7. The last Section 8 concludes this review paper.

## 2 Introduction to object detection

Different types of Object Detection Techniques, Datasets USED so far, and Performance Analysis are explained briefly in this section. Then in the following Section, various Face mask detection algorithms are thoroughly discussed.

**Object detection** is a branch of computer science related to computer vision and image processing [63]. Image Processing is one of the fastest-growing technologies today. In image processing, we can do Image Classification, where we can simply give output by assigning labels to an image. In contrast, in Image Localization, we are finding where a (single) object exists in an image with the help of bounding boxes. Object Detection is a combination of both Image Classification and Image Localization in which the goal is to find objects of specific target classes with their localization in a given image and assign a class label to the detected object or Object Detection is concerned with *what is* in the image and *where* it is in the image. From the 1990s to now, Object Detection has been actively studied due to its tremendous applications such as Pedestrian Detection, Face Detection, Text Detection, Traffic Sign and Traffic Light Detection, and Remote Sensing Target Detection [101]. Different Object Detection Techniques have been invented so far, each with its uses and limitations. Object detection algorithms are used not only to detect objects in images but in videos too. Nowadays, these algorithms are widely used in real-world applications such as in surveillance Cameras, autonomous driving, etc.

At first, we have Traditional Object detection Algorithms with built-in shortcomings:-"Sliding window Problems," which is an exhaustive approach to find out all the possible positions of an object in an image, Manual Feature Extraction, and due to occlusions, localization becomes a challenging task. So, to address these problems, Deep Learning Algorithms have been developed that outperform the Traditional Algorithms. Deep convolutional neural networks (DCNN) are being widely used for better image classification having high computational power, and Detection speed has been increased to meet the needs of real-time system applications while maintaining accuracy. Due to Covid-19 Pandemic, Face Mask Detection is a recent area of interest for Researchers. Face Mask detection falls under "Object Detection," where Mask is treated as an object, and the main task is to detect the mask in an image

### 2.1 Object detection methods

Object Detection methods have been classified into two categories that follow **Non-Neural approaches** (Traditional Object Detection Algorithms) and **Neural Network Approaches** (Deep Learning-Based) (Fig. 4).

### 2.2 Non-neural object detection

The Problem Statement of any Object Detection Algorithm is to find the location of an object in an image **(Object Localization)**, and then we have to classify that particular object into

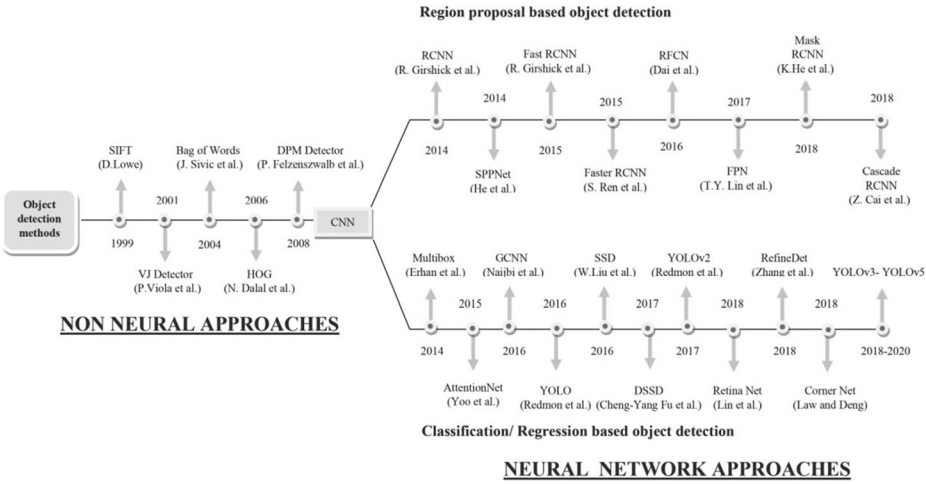



**Fig. 4** Chronological Diagram of Object Detection Methods.

different categories **(Classification)** so to solve this Problem earlier, the pipeline of Non-Neural object detection methods primarily consisted of four stages as shown in Fig. 5.

1) **Informative Region Selection**: In this stage, the main aim is to find the **Regions of Interest (ROI)** that are the locations in an image that contains Objects. An image may consist of multiple objects at different locations that may vary in size or aspect ratios. The idea is to scan the whole **Input Image with** the help of the **"Multi-Scale Sliding Window"** to find out the respective Locations. However, the Sliding Window approach has its limitations; it may be difficult to capture every part of an image and find out all the positions of an object; due to this, it is a computationally expensive process and may produce redundant windows. To overcome this, we can fix the number of windows, but we may miss some critical regions and end with Inappropriate Results with unsatisfactory proposal generation.

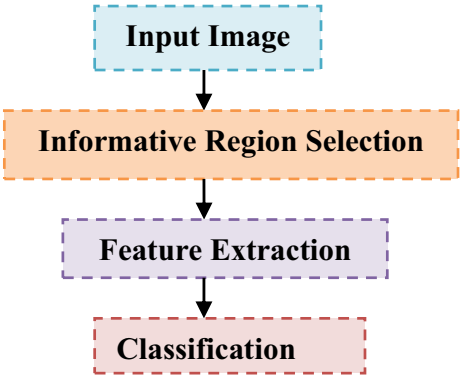



**Fig. 5** The fundamental framework for Non-Neural object detection algorithms

2) **Feature Extraction**: Initially, Feature Extraction Process was done manually. For Object Recognition, Visual Feature Extraction can be done to provide a semantic and Robust Representation of an object. In the previous step, the Sliding window gives a fixed-length feature vector for each location of an image and to encode that the feature extractor uses various visual descriptors such as Scale-invariant feature transform(SIFT) [55], Haar-like features [48], histograms of oriented gradients (HOG) [19], and SURF (Speeded Up Robust Features) [6]. Nevertheless, it is still challenging to design a robust model that can adequately recognize different objects because of its variability in the appearance of an object due to illumination, lighting conditions, noise, scale, occlusion, and background.
3) **Classification**: In this stage, we try to categorize the target object from all the other classes and assign labels to it. Different classification techniques were used, such as support vector machine (SVM) [15], AdaBoost [30], and deformable part-based model (DPM) [29], bagging [64], cascade learning [89], to make the representation more semantic, informative and hierarchical for visual recognition.

A. NON-NEURAL APPROACHES (TRADITIONAL OBJECT DETECTION): There are different Non-Neural approaches that are explained as below:-

1) **Voila-Jones Object Detection (VJ Detector):** *P. Viola* and M. Jones introduced one of the popular real-time Object Detection Framework in 2001 [88], which was later referred to as "Voila-Jones Object Detector"**(VJ Detector)**. The Viola-Jones Object Detection Framework can detect objects in images rapidly and accurately, but it was mainly designed for "Human Face Detection". VJ Detector can process a 384 × 288pixel image in just .067 seconds approximately on a 700 MHz Pentium 3 Processor [88], which implies that the detection is high-speed, but on the other hand, Training time is very slow. It is very Robust in nature, having a high detection rate (true-positive rate) &a very low false-positive rate [90]. The Voila-Jones Algorithm is divided into Four significant Steps:

   i. **Haar Feature Selection:** Alfred Haar, a mathematician, proposed Haar wavelets in 1909 [1]. The Haar-like Features were developed by Paul Viola and Michael Jones by adapting the idea of Haar wavelets. These are the rectangular regions consisting of pixels masked over an image. Within each Rectangle, the summation of pixels is calculated, and then the difference between the Shaded and unshaded regions is calculated, resulting in a single value, say delta!. The most common types of Haar features are **Edge Features** and **Line Features**. While detecting a part of the Human Face, **Edge Features** are more suitable for the Eyebrow region (Shaded Region) as it will be darker and the skin (unshaded Region) is on the lighter side [90], whereas **Line Features** are used for shapes of lips region going from dark-light-dark regions or for a nose as middle part is lighter surrounded by two darker regions. Similarly, scan the whole image for each feature type and then calculate the delta values that will be used further in AdaBoost Training.
   ii. **Creating an Integral Image:** The main goal is to reduce the processing time. In a 24 × 24 pixel image, there are about 160,000 potential feature combinations**,** so addition and subtraction for all the features are computationally heavy. To get rid of this problem "Integral Image Representation" Concept takes place in which the pixels above and left of the corresponding pixel in the source image are added to each point in the integral image. So, Instead of making additions for all pixel values for all

features, we utilize an integral image to achieve the same result with a few subtractions. The speed is automatically increased hence reducing the calculations of all the pixels as now only four corner values are considered of the Rectangle in an integral image.

iii. **AdaBoost (Adaptive boosting) Training/Learning:** As we know, for a 24 × 24 pixel image, there are about 160,000 potential feature combinations that may or may not be useful, and the main aim is to get only the useful features by eliminating the useless ones to get more accurate results, and for that, we have AdaBoost (Adaptive Boosting) Algorithm which is a machine learning algorithm that selects the valuable subset of features from a large number of features. In this algorithm, for each feature, one classifier is created, and each one of these classifiers is known as **Weak Classifiers**, which is then combined with their respective weights to form a **Strong Classifier** which is the output of the AdaBoost Algorithm. After the completion of training, the error rate is calculated, and with the help of this, we can find the best weak classifiers based on some threshold value, and accordingly, valuable features are kept, and useless classifiers are dropped.

$$F(x) = w_1 f_1(x) + w_2 f_2(x) + \ldots..$$
$$w_1, w_2 : \text{weights}$$
$$f(x) : \text{weak classifiers}$$
$$F(x) : \text{Strong Classifier}$$

iv. **Cascading Classifiers**: After Performing AdaBoost Training, we get almost 25,000 features, and it still requires extensive computation, So to increase the speed and accuracy of our model, a set of classifiers **(F1, F2...)** are applied to each sub-window. In this, the first classifier **(F1)** will discard the negative sub-windows whereby accepting only positive results; similarly, subsequent layers perform computations and accept and reject the sub-windows according to their outcome. A negative outcome at any stage rejects the sub-window immediately, which will result in the reduction of negative sub-windows radically that, in turn, boosts the model's speed and helps in real-time face detection [88] (Figs. 6 and 7).

2) Histograms of Oriented Gradients(HOG): Histogram of Oriented Gradients is one of the feature extraction techniques in computer vision and image processing to detect objects. It was proposed by N. Dalal and B.Triggs in 2005 [19]. This descriptor is designed to be computed on a dense grid of uniformly spaced cells, and for better performance, it uses overlapping local contrast normalization [101]. HOG descriptor can be used for different object class detection, but it was mainly designed for "pedestrian detection" [79, 101]. The HOG descriptor focuses on the shape or the structure of an object [62, 76]. For HOG Feature Vector (O/P), the image is broken down into cells, and for each cell, we calculate

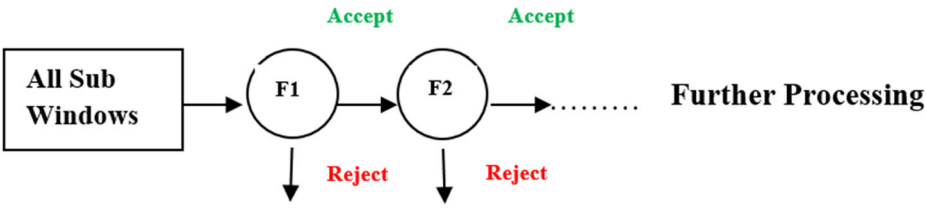



**Fig. 6** Cascading Process

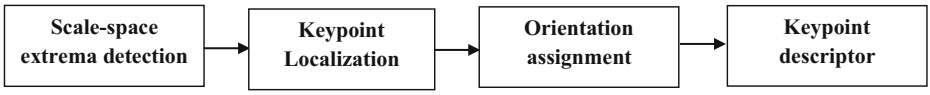



**Fig. 7** The algorithm to obtain the set of features from an image consists of four steps [88]

the gradient (change in X and Y direction), then we determine the Gradient Magnitude and Orientation (Direction/Angle) using mathematical Calculations, after that, we Create Histograms using Gradients and Orientation. In the end, local normalization of cell histograms is performed due to the variability in the images, which in turn helps to enhance the accuracy [25, 79].

3) Scale-invariant feature transform (SIFT): SIFT is a low-level visual descriptor that is used to encode the fixed-length feature vector obtained from the sliding window at each position of the image. It was proposed by David Lowe in 1999 [55]. The SIFT descriptor is particularly beneficial for image matching and object detection. This technique is used to extract features that are invariant to several transformations such as Translation, Scaling, Rotation, and robust to changes in illumination, affine distortion, and noise addition [56]. This image descriptor has various applications, such as view matching for 3D reconstruction, Robot localization and mapping, and human action recognition [56, 77].
4) **Deformable Part Based Model (DPM):** DPM was initially proposed by P. Felzenszwalbin 2008 [26], and later on, R. Girshick made several enhancements to the DPM Detector to deal with variations in real-world objects. For this work, they were awarded the "lifetime achievement" by PASCAL VOC in 2010. DPM detector comprises a Root filter (equal to HOG detector) and the number of part filters and is considered as the extension of Dalal and Triggs Detector (HOG Detector). HOG Detector was only dealing with partial occlusions with fewer variations, and for non-rigid bodies, this was an essential concern because the human body is deformable in nature; one can move their arms, and legs independently, unlike rigid bodies (e.g., sofa, car, Bicycle). DPM Detector follows the "**Divide and Conquer**" Strategy, where training can simply be thought of as learning a proper way to decompose an object, and the inference can be considered as a collection of detections on various parts of objects. For example, the problem of **"face detection"** where the root filter only captures the face boundary, but it can be considered as detection of its parts such as nose, mouth, eyes, etc. using part filters where all configurations of part filters instead of manually can be learned automatically by using weakly supervised learning Technique. DPM Detector is also the winner of VOC-07, −08, and − 09 detection challenges.

There are other non-neural approaches such as **SURF** (Speeded Up Robust Features) [6] and **Bag of Words**.

### 2.3 Deep learning-based object detection: A brief history

Before moving on to Neural Network based Object Detection Approaches, This section begins with the brief history of Deep Learning based Object Detection, why deep learning-based object detection methods over conventional handcrafted feature-based methods (Non-Neural Approaches)?, and an introduction to the most representative deep learning models that are **Convolutional Neural Networks (CNNs)** along with its basic architecture.

Deep Learning is a subset of Machine Learning that is based on **Artificial neural networks**(ANNs) that were introduced in the 1940s [68] to solve learning problems by simulating the Human Brain(Biological Neuron). It consists of an Input layer, Hidden layers, and an output layer. During the processing of the hidden layers, the input features get multiplied with corresponding random weights along with bias. Then, after that, some non-linear functions (Activation functions) are applied to get the desired output. Hinton et al. [75] developed the back-propagation algorithm in the late 1980s and 1990sin which error is computed, and based on some threshold value, weights are adjusted by back-propagating in the network to achieve predicted output. With respect to **Object Detection**, there was less growth between 2010 and 2012, and only minimal gains were obtained by developing ensemble systems and using some minor variants of traditional methods, but after the regeneration of CNN in 2012 [44], due to its low level to high-level feature extraction capability and robustness, researchers shifted their focus to CNNs and later on from 2014 [34] with the introduction of RCNN, object detection began to evolve.

In Non-neural approaches, features were hand-engineered or were manually designed, whereas because of the deep architecture of deep convolutional neural networks in neural approaches, it can learn more complex features and hence widely used in challenging problems nowadays.

Neural network approaches are commonly based on **Convolutional Neural Networks (CNN or ConvNet)**.

It is a deep Learning-based Algorithm that has been widely used in Computer Vision Problems for Feature Extraction and Image Classification. ConvNets are partially connected multilayer Networks which means each neuron in a layer is connected to some of the neurons in the next layer; hence it results in Parameter Reduction and boosts up the Training of the model, and due to this, CNN overpowered Fully connected feedforward neural networks [2]. As shown in Fig. 8**.** CNN takes an Image as an Input(**Input Layer**), then this image is represented in the form of a 3D matrix of pixel intensities for different color spaces [5]; CNN may consist of a one or more **Convolutional Layers** in which "Convolution Operation" is performed with the help of an element known as **Filters/Kernel-**These Filters shifts according to the value of **Stride/ Shift**(Hyperparameter) in each iteration until the entire image is being processed, the output of this layer is the **Feature Map** which is obtained by calculating the Dot product of the input pixel matrix and the filter. We can have more than one filter. E.g., In **Face Detection**, we have different filters for the nose, eyes, mouth, etc. After applying Convolution Operation, dimensionalities of feature maps may be reduced or increased as compared to an input image, and

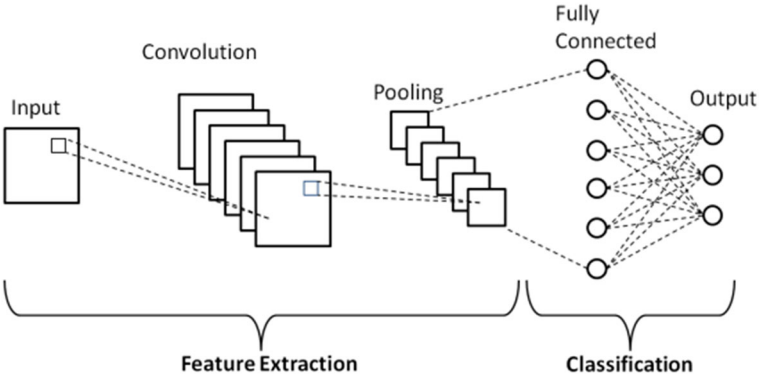



**Fig. 8** Basic convolutional neural network (CNN) architecture [67]

**Padding** is an additional process that can be applied to equalize the dimensions; Convolutional Layer is followed by the **Pooling Layer** in which different pooling operations such as MaxPooling, Average Pooling, L2 pooling, global pooling, etc. are used to reduce the size/pixel/dimensionalities of feature maps (Upsampling/Downsampling of feature Maps) [5]. With the reduction in dimensionalities, computations become much easier [2], and it will be useful to pick the maximum intensity value from the feature map with the help of a pooling operation. Then the high dimensional feature map is flattened into low dimensions and fed into the **Fully connected Layer** in which the actual Classification takes place by using activation functions and probability values(0–1) to obtain an **Output** by the assignment of input features in the valid classes. CNN has many applications in Object Detection, Image Classification, Image Segmentation, medical image analysis, natural language processing, etc. [14].

## B. NEURAL NETWORK APPROACHES (DEEP LEARNING-BASED OBJECT DETECTION):

Nowadays, deep learning-based object detection frameworks can be categorized into two groups:

1) **Region Proposal Based Object Detection (Two-Stage Detectors):** In Region Proposal based framework, it is a multi-stage process in which, in the Region Proposal Generation stage, we try to generate region proposals by adopting the "Selective Search" approach in which, unlike the traditional based approach where the multi-scale sliding window is used and slide over the entire image including the non-interesting regions, here we find a region of interest(ROIs) by selecting only that regions which may potentially consist of an object, in Feature Extraction with CNN stage [44], from the generated regions we extract the features using CNN and finally in classification and Localization stage we assign the labels to the proposed region according to their predicted class and localize them by drawing the bounding boxes around it.

   - **RCNN**: In 2014, R. Girshick et al. [85] introduced an Object Detection Model, i.e., RCNN(Regions with CNN features), which was a progressive step as it achieved more than 30% improvement on PASCAL VOC 2012 over the previous results. As shown in Fig. 9**.** RCNN model consists of three modules; first, the **Generation of region proposal** where from an **Input image** by adopting the "Selective Search

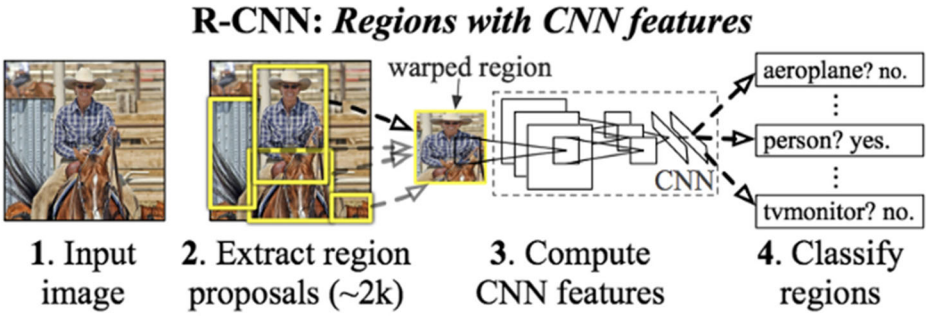



**Fig. 9** Architecture of RCNN [34]

Approach [85]" in which based on pixel intensities, regions were grouped, about 2000 category-independent regions/region proposals per image were extracted/generated, after that before passing it to CNN(Alexnet) **Warping** or **Cropping** is performed in which each region proposal was resized into a fixed 224 × 224 pixel size **(warped region),** then in the second module, **Feature Extraction** is performed by using CNN to extract 4096-dimensional feature vectors for each region proposal [45, 47, and in the third module, **Classification** is performed by assigning scores to each extracted feature vector using pre-trained linear SVMs to predict the actual presence of an object and an additional step Bounding Box Regression and Non-Maximum Suppression is performed to precisely locate that object by drawing a bounding box. *Even though RCNN has made significant progress over Non-Neural approaches still, it has some limitations that are described as below:*

i. It generates about 2000 region proposals for an image, and there may be some overlapping proposals on which redundant computations are performed, making it an exhaustive approach [23].
ii. Training is a multi-stage pipeline; hence it is very slow to implement [33].
iii. Expensive space and time for training [33].
iv. We need to resize each region proposal (2 k) to a fixed size input for CNN, which would automatically increase its testing time.
v. In this model, CNN is repeatedly applied to 2000 regions which is a time-consuming process [58].
vi. The cropping or Warping Process may result in loss of content or geometric distortions, respectively, which in turn affect the accuracy as well [5].

- **SPPNet:** Later in the same year in 2014 in which RCNN was proposed, K. He et al. came up with the idea of **Spatial Pyramid Pooling Networks**(SPPNet) [58] to overcome fixed input image size constraint (224 × 224) of RCNN(fully connected layers) where at each region Cropping, or Warping Process is performed which may result in loss of content or geometric distortions respectively [5]. In SPPNet, an image is given as input, then CNN extracts the feature map from an entire image only once, which saves much time rather than applying CNN 2000-times on each region as in RCNN and then as shown in Fig. 10. they introduced the **"spatial pooling layer"**(in between the last convolutional layer ($conv_5$) and fully-connected layers)with multiple variable scale poolings (combined to form a fixed-length representation) where each feature map generated from the $Conv_5$(last convolutional layer) is combined into a single value, four values, 16 values resulting in a 256-d vector, 4 × 256-d vector, 16 × 256-d vector respectively to extract the feature vectors of fixed length on the feature map and after that this fixed length Representation is fed into the fully connected layers($fc_6$,$fc_7$).*SPPNet is significantly faster than RCNN while maintaining the accuracy of detection (VOC07 mAP = 59.2%), but still, it has some drawbacks that are described as follows* [101]*:*

  i. The training continues to be multi-staged.
  ii. SPPNet does not take care of the previous layers of the network as it is only concerned about the fully connected layers.

- **Fast RCNN:** In 2015, R. Girshick [33] proposed a fast version of RCNN for Object Detection, i.e., named as ***"Fast RCNN".*** Even SPPNet improves RCNN

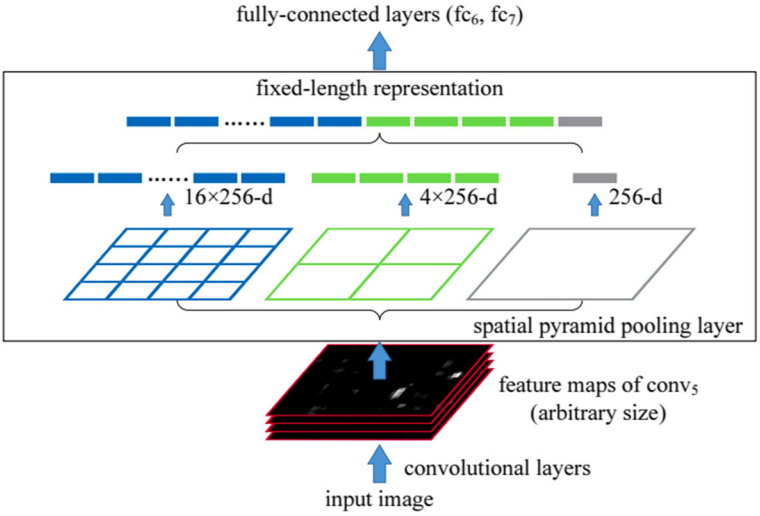



**Fig. 10** Architecture of SPPNet with 256 filters in theconv5 layer (last layer) [58]

in many ways still it has some notable drawbacks like the problem of multi-stage training (Feature Extraction stage, Fine-Tuning stage, SVMs Training, Bounding Box Regression stage),it do not update the convolutional layers, Storage-space issues that were later solved by an introduction of **Fast RCNN Architecture** that is shown in Fig. 11**.** in which similarly like SPPNet, Fast RCNN extracts features from an entire image to generate a **ConV feature map** then Spatial pyramid pooling layer in SPP-net is replaced by ROI(Region Of Interest)pooling Layer that extracts a fixed length feature vector from the Feature Map for each Region proposal generated by applying "Selection Search Algorithm" then **feature vectors** given as an input to the fully connected layers**(FCs)** and at the end they fed into two sibling output layers: one that computes **SoftMax probabilities** of C + 1(C denotes "object classes" and plus 1 for "Background class") and the other one is for **Bounding Box Regression**(having 4 real valued coordinates for each Object class C),So with the multi task loss this architecture is end to end trainable and it jointly performs the Classification and BB Regression by sharing convolutional

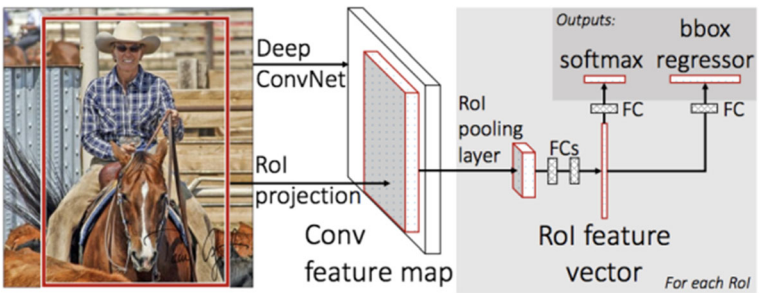



**Fig. 11** Architecture of Fast RCNN [33]

features instead of independently training SVMs and BB Regressors as in SPPNet/RCNN.

*In comparison to RCNN/SPPNet, Fast RCNN has the following advantages:*

i. On PASCAL VOC 2007 dataset, mAP is increased from 66.0% (RCNN) to 66.9% in Fast RCNN [23], so the detection Quality (mAP) is comparatively High.
ii. End-to-end Training using multi-task loss.
iii. All the layers get updated in this network.
iv. For feature Caching, there is no extra storage space required hence removing the memory constraints,
v. Speed is automatically accelerated due to fast training and testing procedures.

- **Faster RCNN**: Even-though Fast RCNN has shown tremendous results in terms of detection accuracy and speed still, the computation of region proposals by using the traditional region proposal algorithm, i.e., "Selection Search Algorithm [85]" in which about 2000 region proposals were generated from an image and then featured maps were extracted that becomes a bottleneck of the object detection architecture because it is a time-consuming process [20]. Therefore, to solve this problem, In 2015, S. Ren et al. [73] proposed the **Faster RCNN** detector in which, for the generation of region proposals, he replaced the selection search algorithm with the **Region Proposal Network(RPN).** Figure 12(a)**:** shows the Architecture of Faster RCNN**," Faster RCNN** is a combination of **RPN** and **FAST RCNN,"** where an entire image is given as an input to the deep CNN (conv layers) to generate feature maps that are shared between the RPN and the detection network with some last convolutional layers. Figure 12(b): shows Region Proposal Network (RPN), where the conv feature map acts as an input to the RPN and **anchor boxes** are the output generated by the sliding window. In this, n × n(n = 3) spatial window slides over the conv feature map, then a low dimensional vector is generated(256-d), which is then fed into two output sibling layers: ***cls layer***(Classification layer) and ***reg layer***(Regression layer) that provides 2 k scores(probability estimates of the presence of object for each proposal) and 4 k coordinates (bounding box coordinates)

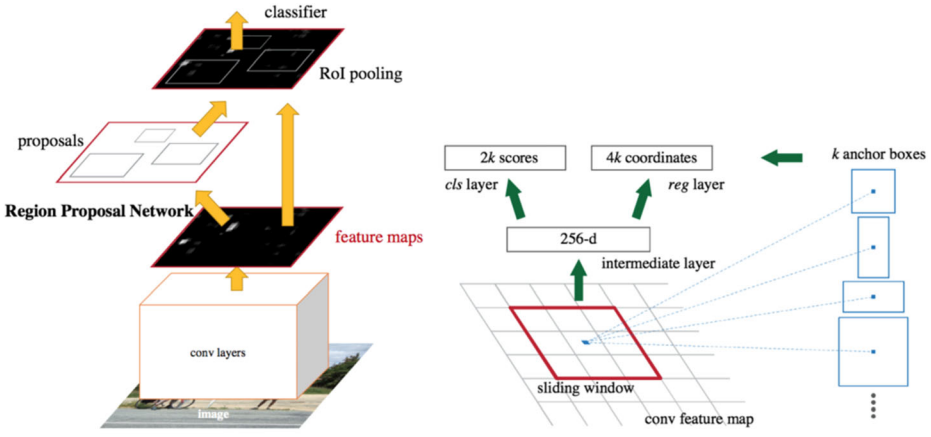



**Fig. 12** (a)Architecture of Faster RCNN. (b)Region Proposal Network (RPN) [73]

respectively where k(k = 9) is maximum possible regions **(anchor boxes)** for each sliding window location. Then the generated anchor boxes/regions are finally fed into the ROI pooling layer for the fixed-size representation, followed by classifier and regressor as in Fast RCNN.

**Advantages**:

i. On PASCAL VOC 2007 test set, mAP is increased from 66.9% (Fast RCNN) to 69.9% in Faster RCNN [23].
ii. RPN is an accurate and efficient region proposal method.
iii. Faster RCNN decreased the number of proposals from 2000 [20].

- **RFCN (Region-based Fully Convolutional Network):** Although traditional Region proposal networks such as RCNN, Fast RCNN, and Faster RCNN have achieved significant improvements by sharing the feature extraction computation for various Region proposals but still in RoI-wise sub-network, each region proposal(*may be hundreds per image*) goes separately to the sequence of fully connected layers which is a time-consuming process and makes the network slow. So, to address this issue, for the detection of objects more accurately, efficiently, and fastly, in 2016, Dai et al. [18] proposed RFCN (Region-based Fully Convolutional Network). As shown in Fig. 13(a): Similarly to Faster RCNN, ROIs are extracted by RPN, but the fully connected layers after the ROI pooling layer were removed and in Fig. 13(b), Image is given as input, and ResNet-101 is used by RFCN for feature extraction, where after the generation of feature maps they apply $k^2(C + 1)$-d convolution (C: "object classes," plus 1: "Background class," $k^2$ are the feature maps for each class) to create "position-sensitive score maps". For e.g., If we take k = 3, then a total of 9 score maps were created (top-left TL, top-center TC, top-right TR, bottom-right BR). Then in the position-sensitive ROI pool, these score maps and ROIs are mapped to the vote array(k × k). In the end, the average of this vote array(Average Voting) is computed to generate the class score for each class, and then a softmax classifier is applied to generate the class probabilities, followed by Bounding Box Regression. On PASCAL VOC 2007 datasets RFCN achieves mAP = 83.6%.
- **FPN:** Previous works such as **Faster RCNN** use only a Single scale feature map to obtain the final predictions; hence detection of small objects with different scales becomes a challenging issue. In deep ConvNet, a feature hierarchy is obtained corresponding to each layer where the deeper layers' feature maps have essential information as they are *semantically strong. Still,* they are *spatially weak,* having low resolutions, similarly shallow layer features are *spatially robust* having high resolutions, but they are *semantically weak* that degrade the detection accuracy [94]. So, In

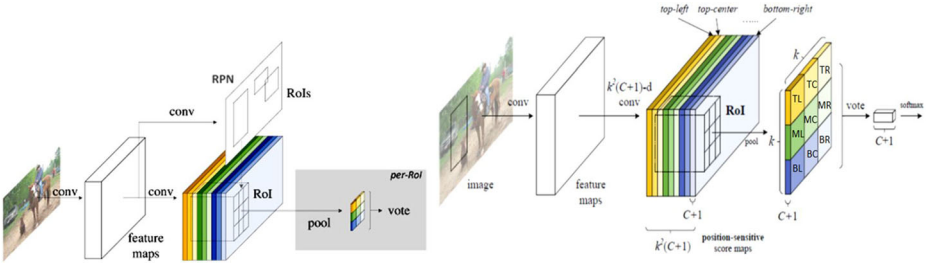
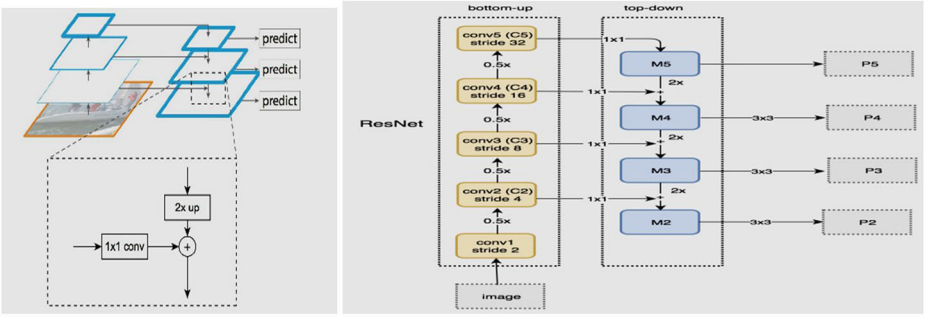



**Fig. 13** Architecture of R-FCN [18]

2017, Lin et al. [49],introduced FPN (Feature Pyramid Networks) which is a Feature Extractor commonly used in object detection that fuses both spatially robust shallow level feature maps with semantically strong deep level feature maps in order to obtain multi-scale feature maps that significantly improve detection performance via Top-down pathway and lateral connections as shown in Fig. 14(a) whereas Fig. 14(b). Represents the detailed view of FPN which includes Bottom-up Pathway, Top-down pathway, and lateral connections. An input image is given to the Bottom-up pathway, which is a feed-forward Convolutional neural network (Architecture used: ResNet) having different convolutional layers (conv*i* where $i$ = 1to5) that generates the feature maps at different scales with scaling step = 2(doubling the stride and reducing the dimensions by 0.5), then the outputs of conv*i* denoted as C*i* are applied to a convolutional layer having filter size = 1 × 1 to have fixed number of channel dimensions(by default:256-d) will merge to respective feature maps of M*i* using element-wise addition (Conv layers) in Top-down Pathway and for the addition of two feature maps the channels should be identical in ResNet. Finally, a 3 × 3 convolution filter is applied to merged maps (M*i*) in order to get all pyramid feature maps(P2, P3, P4, P5with 256-d output channels)and to reduce the aliasing effect of upsampling; C1 is not considered here due to memory constraint [86].

- **Mask RCNN:** In 2017, He et al. [36] proposed a general and straightforward framework known as Mask RCNN to solve the "Instance Segmentation" problem, which further consists of two sub-problems such as Object Detection(detecting and classifying the object in an image) and Semantic Segmentation(Pixel-level Image Understanding). Mask RCNN is an extension of Faster RCNN in which an additional branch ("mask branch") is presented for a pixel-to-pixel segmentation mask prediction in parallel to the two existing branches for classification and bounding box regression. Mask RCNN for Object detection works in a similar manner as Faster RCNN by employing RPN in its first stage, and for semantic segmentation, it uses Fully Convolutional Network (FPN) for the prediction of m × m mask from each region. ROI pooling layer in Faster RCNN consists of Quantization operation (stride is quantized) is performed in which, e.g., We are having 16 × 16 RoI and to map it to 7 × 7 space we have a stride = 2.28(16/7) and then round off operation is performed to convert floating number into integer representation, as a result of these quantization's, it causes misalignments between ROIs and Features and loss of information. To handle such things, they propose the ROI Align layer as shown in




(a) (b)

**Fig. 14** (a)Architecture of FPN [49] (b) Data Flow [86]

Fig. 15, which avoids quantization (16/7 = 2.28) and uses bilinear interpolation for pixel-level alignment. It is easy to implement, flexible in nature, and trains with a less computational overhead of the mask branch. Mask RCNN achieved more significant improvements and accuracies over Faster RCNN, and it got the championship of the COCO 2016 object detection challenge [20].

- **Chained Cascade Network and Cascade RCNN:** To evaluate the complex classifier on an entire image, classifiers are divided into a *"Cascade"* (Linear sequence) of sub-classifiers to reduce the computational overhead [8]. The importance of cascade is learning more discriminative classifiers using multi-stage classifiers. It rejects many negative samples at earlier stages so that classifiers in the following layers can handle more difficult examples [66]. **Two-stage detectors** for object detection also follow the cascade approach in which, at earlier stages, background samples are removed for better learning of classifiers, and in later stages, the remaining regions (ROIs) are used for classification. The Cascades have been widely used for object detection [8, 27, 47] because they increase the detection process's accuracy and speed by removing simple background samples during both training and testing [66]. A **chained cascade network** was presented [66] for object detection in which a single end-to-end neural network is used to learn the cascaded classifier's multiple stages, which was later extended in **Cascaded RCNN** [11]. Cascade architecture can also be adopted to solve the **Face detection problem,** in which non-faces are rejected at earlier stages and faces are passed on to the following stages [8]. Recently, Hybrid Task Cascade(cascade architecture) has been proposed, for instance, segmentation also, ranking 1st in the COCO 2018 Challenge Object Detection [12].

2) **Classification/Regression-Based Object Detection (One-Stage Detectors):** Region Proposal-based Object detection methods consist of multiple stages such as Region proposal generation, feature extraction with CNN, classification, and bounding box regression that are generally trained separately. In real-time applications managing these different stages are also not possible due to its time limitations. These methods are also incompatible with mobile/wearable devices because of their computational expense and memory constraints. Now, researchers turned their focus to One-stage detectors because it is a one-stage process as it does not involve a Region proposal generation stage like the RCNN family; it simply predicts the class probabilities and bounding box coordinates

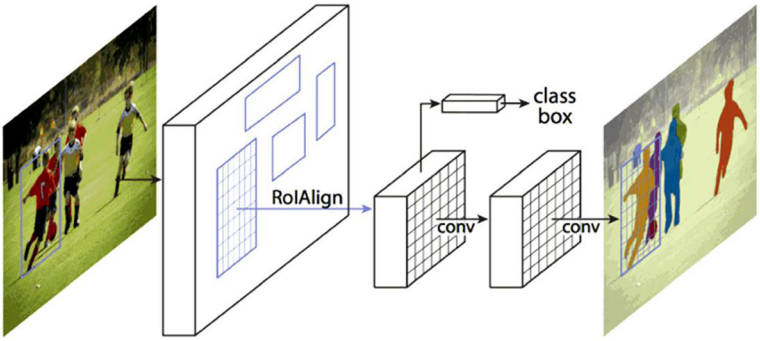



**Fig. 15** Mask R-CNN framework for instance segmentation [36]

directly from the entire image by considering each location of an image as potential objects. These detectors are comparatively fast *and more straightforward, even though they may slow down the performance slightly*, but they have been widely used in real-time applications and also help reduce the time expense. Firstly, we look at various early CNN models before focusing on other vital frameworks such as YOLO [72] and SSD [52].

- **Pioneer Works:** Before popular one-stage detectors such as YOLO and SSD, a lot of work has been done to improve object detection models as regression/ classification tasks.

  i. **Multibox**: In 2014, Erhan et al. [22] and Szegedy et al. [80] proposed the Bounding box regression Technique in which a "deep neural network" is used to generate the bounding boxes in a class-agnostic manner, and for the generated box, it also outputs the confidence score with it that describes the presence of an object in that box". Loss" was introduced to bias both localization and confidences of various components to predict the class-agnostic BBs coordinates [100], as well as various contributions, have been made to the last layer.

  ii. **AttentionNet**: In 2015,Yoo et al. [97] considered an object detection problem an iterative classification problem and introduced an end-to-end method using a deep convolutional neural network(CNN) named **AttentionNet**. This network generates quantized weak directions that point to a target object beginning from the top-left and bottom-right corners of an image, and the network converges to an accurate estimation of the object bounding box with an ensemble of iterative predictions. AttentionNet is a unified network as it does not consist of separate stages such as object proposal, object classification, and bounding box regression. AttentionNet may give us impressive results as it achieves AP = 65% on PASCAL VOC 2007/2012 with an 8-layered architecture, but it is not scalable to multiple classes, and it has a low recall.

  iii. **G-CNN:** In 2016,Naijbi et al. [60] developed CNN-based object detection technique, i.e., GCNN. It is an iterative grid-based object detector that has a "no object proposal stage. G-CNN models the object detection problem as finding a path from a fixed grid to a bounding box fitting a target object. It begins with a fixed multi-scale bounding box grid over an input image, and then a regressor is trained to move repeatedly and scale the grid elements towards objects [60]. GCNN is five times faster than fast RCNN, but it does not work well for small or highly overlapping objects.

- **YOLO:** In 2015, R. Joseph et al. proposed the first one-stage detector during the deep learning period, i.e**., YOLO(You Only Look Once),** as you only look once (YOLO) at an input image to predict *what* objects are present and *where* they are [72]**.** It is a unified model that considers object detection as a regression problem as it directly predicts the bounding box coordinates and class probabilities from the entire input image in one go, thus called an "end-to-end single neural network". As shown in Fig. 16**,** an input image is fed through the CNN architecture (GoogLeNet), which consists of 24 convolutional layers followed by two fully connected layers with 1 × 1 reduction layers to reduce the feature space from previous layers, then the **YOLO Algorithm** works by dividing the image into S × S grid and particularly each grid

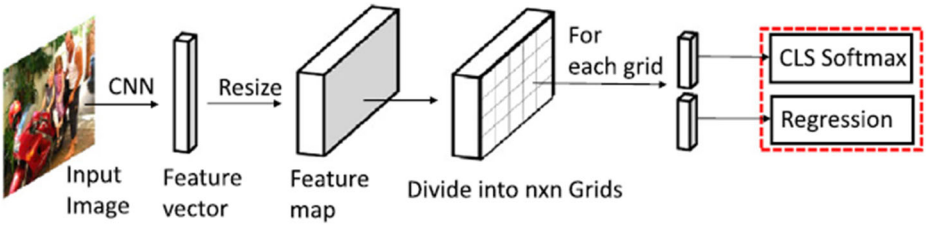



**Fig. 16** Architecture of YOLO [94]

cell having a center of the object in it is responsible for object detection. Each grid cell predicts **bounding boxes coordinates(B) [(x, y): Center of the box, w: width, h: height],** associated **confidence scores**(how confident the model is about the presence of an object in the box and how accurately the box predicts the boundaries of the object) and **Conditional class probabilities (C)**and these predictions are encoded as an S × S × (B × 5 + C) tensor [66]. Fast YOLO runs at 155fps with VOC07 mAP = 52.7%, whereas its new version YOLO runs at 45 fps with VOC07 mAP = 63.4% and on the VOC 2012 test set, YOLO achieves 57.9% mAP.

- **Variants of YOLO:** YOLO [72], YOLOV2/YOLO9000 [70], YOLO V3 [71], YOLO V4 [7], YOLO V5 [96].

– Features:

i. It is a state-of-the-art object detection algorithm that is computationally very fast and widely used in real-time environments.
ii. It globally processes the entire image at once with a single forward pass network.
iii. YOLO learns generalizable representations of objects, which also works well for new domains and unexpected inputs.

– Limitations:

i. YOLO does not perform well for small dense objects such as a flock of birds since every grid cell predicts only two bounding boxes, and it can only have one class.
ii. The generalization ability is not much good as it is not suitable for predicting objects at new/ unusual aspect ratios or configurations.
iii. The drawback of the loss function affects the detection performance [20].
iv. Although YOLO boosts up the speed, it lags in terms of accuracy. As compared to Fast R-CNN (mAP = 70.0%, 0.5fps) and Faster R-CNN (mAP =73.2%, 7fps), YOLO achieved 63.4% mAP with 45 fps.

- **SSD:** Despite YOLO's high detection speed, the object generalization ability is still weak as it is challenging to deal with different aspect ratios and scales, and the detection effect for small objects was also limited. To address these limitations of YOLO, In 2016, Liu et al. [52] proposed a model named SSD (**Single Shot MultiBox Detector),** in which an image with ground truth boxes is given as an input to the base model(VGG16), followed by different convolutional feature layers with gradually decreasing in size that allow predictions of detections(default boxes offset) at multiple scales and aspect ratios with their associated confidence scores [52]. Then, a "matching strategy" is performed to train the network to determine the

appropriate default boxes that correspond to the ground truth boxes. The weighted sum of localization loss (e.g., Smooth L1) and confidence loss (e.g., Softmax) is the **model loss**. To eliminate redundant predictions pointing to the same object, SSD employs **a non-maximum suppression** process and produces the final detections. It is faster and more efficient than YOLO as SSD300 achieves mAP = 74.3% at 59 fps on the VOC2007 test.

**DSSD (**Deconvolutional Single Shot Detector) and **FSSD** (2017): These models were introduced as an enhancement over SSD in which DSSD express low-level feature maps and uses ResNet101 as a base model, whereas FSSD combines low-level features into high-level features based on SSD, which significantly increases the accuracy [20].

- **RetinaNet:** During the training of dense detectors**,** there is a class imbalance problem between foreground-background, and to overcome this problem**,** Lin et al. [50] proposed a one-stage detector **RetinaNet** in **2018** by reshaping the standard cross-entropy loss. They introduced **"focal loss,"** a new loss function that focuses on complex training examples and avoids many negative samples. As shown in Fig. 18, they employed **feature pyramid networks** to detect multi-scale objects at various levels of feature maps. Due to the introduction of *focal loss*, in terms of accuracy, RetinaNet outperformed all the existing two-stage detectors as well as it is also capable of maintaining the speed of previous one-stage detectors.
- **RefineDet:** Zhang et al. proposed a single-shot-based detector named **"RefineDet"** [98] that is composed of two inter-connected modules that are **Anchor Refinement Module(ARM)** in which negative anchors are discarded to reduce the search space for the classifier along with the adjustment of locations and size of anchors and the **Object Detection Module(ODM)** in which the refined anchors from the first module gives as an input to this module to improve the accuracy of regression and predict the multi-categories label. Both of these modules are connected via a **transfer connection block** that transfers features from **ARM** to **ODM** for better prediction of objects. The whole network is end-to-end trainable and consists of three stages: pre-processing, detection (two inter-connected modules), and NMS [5]. Recent One-stage Detectors such as YOLO and SSD use one-step regression to reach the final output. However, they presented a two-step cascaded regression method for better small object prediction and gave more accurate object locations.
- **CornerNet:** In previous single-stage Detectors, anchor boxes were designed manually [94], so in 2018, Law and Deng proposed an *anchor-free approach* in which they are detecting objects as paired vital points (the top-left corner and the bottom-right corner of the bounding box), named as **CornerNet** [46]. As shown in Fig. 19,

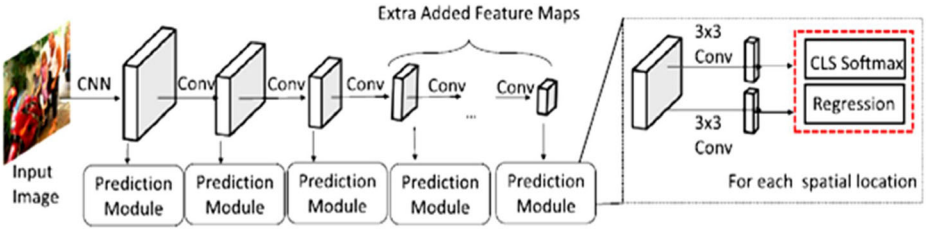



**Fig. 17** Architecture of SSD [94]

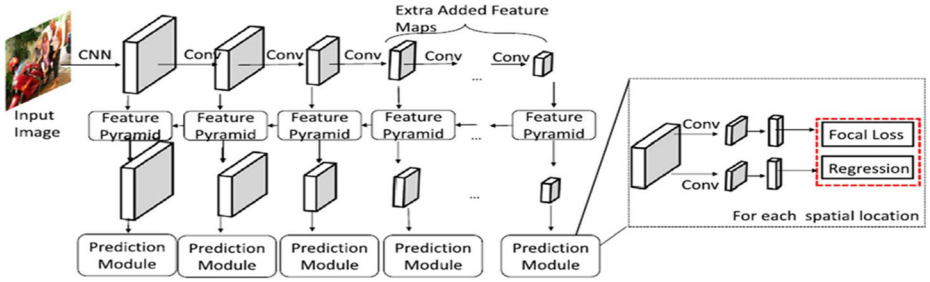



**Fig. 18** Architecture of RetinaNet [94]

The Network predicts **Class Heatmaps** for top-left and bottom-right corners, **Pair Embedding** vector for each corner detected, and These embeddings serve to group a pair of corners that belong to the same objects. **Corner offsets** are predicted for precise corner locations. CornerNet outperforms all previous one-stage detectors such as YOLOv2, SSD, DSSD, RefineDet as it achieves AP = 42.1% on MS COCO Datasets.

In the deep learning era, some other one-stage detectors have been widely used, such as **DSOD(2017), M2det, Efficient-det, and DetectorNet.**

I. *Literature review*:

In 2015,Nieto-Rodr'ıguez et al. [61] proposed a System for Medical Mask Detection that alerts the healthcare workers by triggering an alarm when they do not wear the compulsory surgical mask in the operating room. The main goal of this approach is to reduce the false positive face detections and false alarms rate. In this, the Face Classification system is comprised of four modules: the face detector, the surgical mask detector, two-color filters for face detections, and mask detections. An image enters to "Viola and Jones faces detector" that uses a cascade of classifiers where each classifier is trained via a variant of AdaBoost called LogitBoost and a surgical mask detector that uses Gentle AdaBoost. The detections given by these detectors are in the form of bounding boxes along with a confidence score; after that, these resultant detections passed through respective two color filters that use tone in HSV color space and discard the false positives, and finally, the detections are classified to one of the two classes "Face" or "mask" accordingly. This system has faced various challenges such as due to shades and garment folding, there may be chances of false face detections, incorrectly mask-wearing or

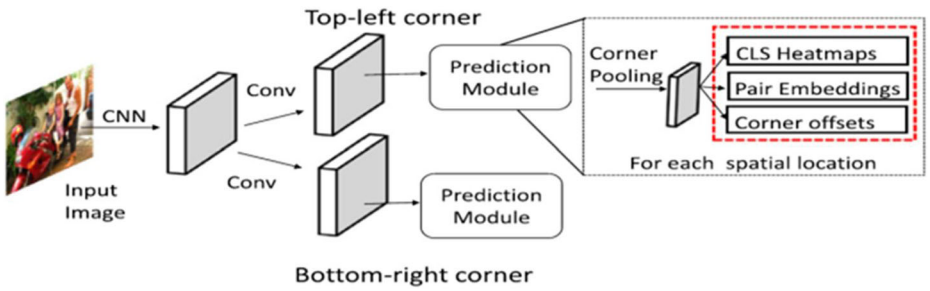



**Fig. 19** Architecture of CornerNet [94]

clothing near to face may give rise to false-positive rate; this system mainly works on surgical masks, so it may not work well with variability in masks such as skin-colored masks, complicated detections having faces with goggles and mask, these issues were solved by employing synthetic rotations. There were difficulties in handling variations in faces (rotated or leaned) which are then solved by increasing frame rates. After the experimental analysis, the recall obtained is above 95%, and a false positive rate is below 5%.

In 2017, Wei et al. [9] introduced the "CNN-based Cascade Approach for Masked Face Detection". This Framework was developed to prevent terrorist attacks by recognition of Terrorists or Criminals. An input image is given to the "masked face detector" having three CNNs such as "Mask-12"(5 layers), "Mask-24-1"(7 layers), and "Mask-24-2"(7 layers) with a classification ability from low to high. Each mask assigned a probability to every detection window, then this probability is compared with the pre-set threshold; if the probability is equal to or above the threshold, then it is accepted for further evaluations; otherwise, it is rejected by the detector. After each Mask, the "Non-maximum suppression" (NMS) technique is used to combine highly overlapped candidate windows. In the end, the "final detection result" was obtained. For the Evaluation purpose, they tested their algorithm on the MASKED FACE testing set and achieved an accuracy = 86.6% and recall = 87.8%.

In 2019, Ejaz et al. [21] implemented an effective statistical technique called Principal Component Analysis (PCA) for masked and non-masked face recognition. This work was mainly presented for security purposes instead of covid-19. In this, they firstly acquired an image from the database; then, in the training phase, they used the" Viola-Jones Algorithm" to detect the face portion from an image; after that, for Facial Feature Extraction, PCA is used. After applying PCA Algorithm, weights are calculated and compared with the test image weight in the testing phase, then; if the difference between both the weights is less than or equal to the threshold value, it is considered as a false detection, and hence face is not recognized else face is recognized. This work concludes that PCA performs better on non-masked face recognition giving an average of 95% accuracy, whereas accuracy dropped to 72% in the case of masked face recognition due to missing facial features.

Roy et al. [74] proposed a real-time face mask detection system based on a deep learning approach. To detect people wearing a mask or not, they have used different object detection models such asYOLOv3, YOLOv3Tiny, SSD, and Faster R-CNN. The system was developed to employ on-edge devices of the surveillance platforms for real-time detections.**Yolov3** [71] is a popular one-stage detector in which an entire image is fed into a single neural network, which results in bounding boxes along with predicted probabilities. "Darknet-53" was used as a backbone network in yolov3. Then they used the lighter form of yolov3 named YOLOv3 Tiny, which works on the same algorithm as yolov3. Still, it has only 2 YOLO layers, and it is comparatively faster due to its low processing overhead, which is important for real-time scenarios. SSD [52] is also a single-stage detector that is used for multiple object detection in an image in a single shot. They fused SSD [52] and MobileNet v2 [37] together because of their simple architecture. Faster RCNN [73] on Inceptionv2fed input image to CNN to generate feature maps which then pass to region proposal network to obtain region proposals then, the classification step is performed, and the bounding boxes are obtained.YOLOv3 608 × 608 achieved mAP = 66.84 at 10.9 fps, YOLOv3Tiny 832 × 832 achieved mAP = 56.57 at 46.5fps, SSD 300 MobileNetv2 achieved mAP = 46.52 at

67.1 fps and F-RCNN 300 Inceptionv2 achieved mAP = 60.5 at 14.8 fps.

Loey et al. [53] proposed a deep learning-based model for medical face mask detection, and this model is comprised of two parts where; for the "feature extraction" part, they used the Residual neural network (ResNet-50), which is a deep transfer learning model that has 16 residual bottleneck blocks, each block has convolution size $1 \times 1$, $3 \times 3$, *and* $1 \times 1$ with feature maps (64, 128, 256, 512, 1024) [53]. Furthermore, for face mask detection, they have used YOLO v2(a few convolutional layers, a transform layer, and an output layer). To increase the detector's performance, the Data augmentation process increased the amount of training data, and they used mean Intersection over Union (IoU) to evaluate the number of anchor boxes. Their proposed detector model has used two optimizer techniques: SGDM and ADAM. The authors reported that the Adam optimizer obtained an average precision = 81% on their proposed model.

Zhang et al. [99] expand the two-class face mask detection problem (correct face mask or without face mask)to the triple-class problem with an addition of a new class(incorrect face mask) and introduced a face mask detection method called as Context-Attention R-CNN about conditions of wearing a mask. Mainly they have implemented a "multiple context feature extractor" for the multiple feature information extraction for different region proposals. Then after employing the "attention module," they decoupled classification-localization branches by separating the parameters for better feature extraction, and finally, a sequence of fully connected layers is used for the prediction of classification scores and localization offsets. Their proposed model outperformed various one-stage detectors, and experiments have shown that they achieved mAP = 84.1% (Figs. 20, 21, 22 and 23).

G. Jignesh et al. [42] developed an automated face mask detection system by using the deep transfer learning-based model of InceptionV3, which is one of the widely used pre-trained state-of-the-art models. Generally, deep learning models need a large amount of training data to perform well, so due to the less number of images in their dataset, they have done oversampling by using the "Image Augmentation" technique in which their dataset was artificially increased by performing eight different operations on the training samples: shearing, contrasting, flipping horizontally, rotating, zooming, blurring. They modified the InceptionV3 Model by replacing the last layer with five new layers: average pooling layer, flattening layer, dense layer, dropout layer, and decisive dense layer. They used the softmax activation function to classify masked and unmasked faces. Their proposed model outperformed various other models, such as MobilenetV2, VGG16, ResNet-50, etc., and has achieved 99.92% and 100% accuracy during the training and testing phase, respectively.

Loey et al. [54] introduced a hybrid framework using the deep transfer learning model"Resnet-50" for feature extraction and classical machine learning techniques such as

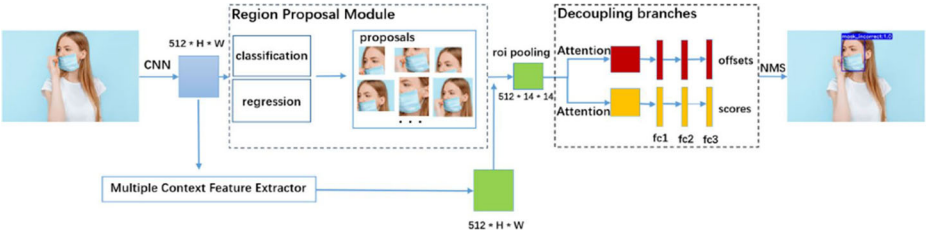



**Fig. 20** Context-Attention R-CNN proposed by Zhang et al. [99]

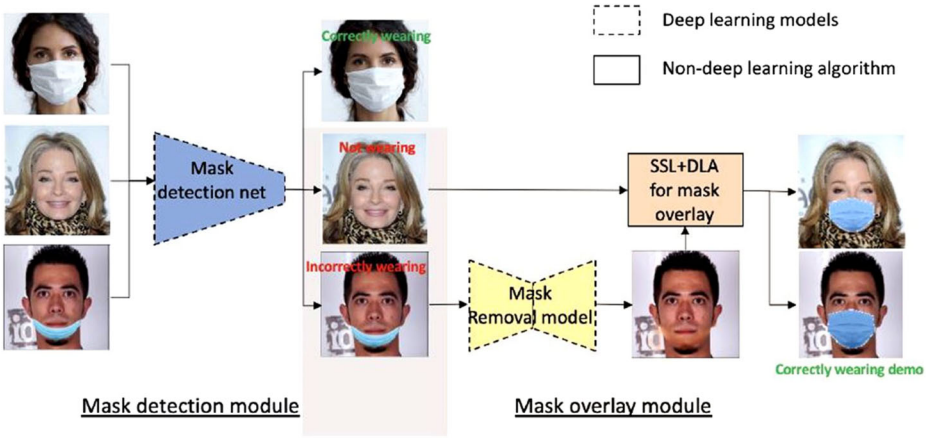



**Fig. 21** The overall Framework of an automated "CoverTheFace" system [38]

"decision trees," "Support Vector Machine" (SVM), and "ensemble algorithms" for Classification. ResNet-50 is based on residual learning having 50 layers: it begins with a convolutional layer, then 16 residual blocks, each having three convolutional layers followed by a fully connected layer, author in their work has replaced the last layer with ML Classifiers such as SVM, decision trees and ensemble methods. After all the experimental analysis, they have noticed that as compared to others, SVM Classifier took less time during the training phase, and its testing accuracy in RMFD, SMFD, and LFW has reached upto99.64%, 99.49%, and 100%, respectively. Although traditional ML Classifiers achieved the highest accuracy, the time consumption is also high as well, and to overcome this issue, they may replace ML Classifiers with various deep learning pre-trained models.

People may wear the mask, but some of them wear it incorrectly, and to address this issue, Yixin Hu et al. [38] developed an automated "CoverTheFace" system in which they detect the images with incorrectly worn face masks, and they provide a visual demonstration on correctly wearing the masks. For this purpose, their system consists of two independent modules: Face mask detection (MobileNetV2) and mask overlay. An input image, when given to the proposed system, the Face mask detection module, will classify it into one of the categories:" correctly wearing," "incorrectly wearing," and "not wearing, where if for the correctly wearing category, the overlay module neglect the further processing but for the incorrectly mask-wearing category, a mask removal model called "GAN-based model(MCGAN)" is used to remove the mask first and then for both the "incorrectly wearing," and "not wearing" categories, an overlay module simply put on the mask on the image but due to different face variations in different images this module also includes "statistical shape analysis(SSA)" and dense landmark alignment(DLA) so that these variations in orientations and face shapes should be well handled. They concluded that their Face mask detection system achieved a detection rate = 98%, whereas the mask overlay module outperformed previous works by employing SSA and DLA strategies.

Zekun et al. [91] proposed a serverless edge-computing design-based face mask detection system named "WearMask" that can be easily deployed on any device and accessible via Web Browsers through a proper internet connection so it automatically reduces the hardware or software costs. Their proposed model employs an edge-computing approach of combining(1)

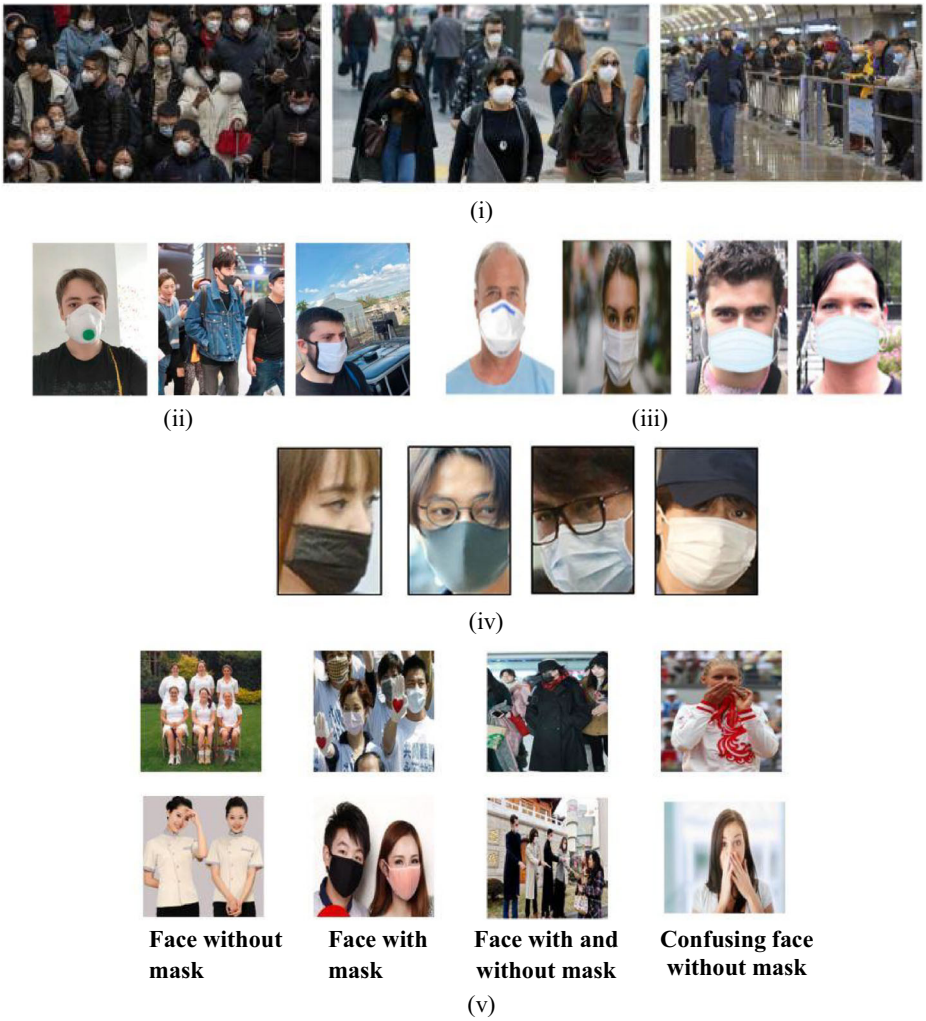



**Fig. 22** Image Samples of different Datasets USED by the authors in existing Literature for Face Mask Detection; i) Image samples of Medical Mask Dataset (MMD) [53], ii) Image samples of Larxel's Face Mask Detection Dataset (FMDD) [53], iii) Image samples of Simulated Masked Face Dataset (SMFD) [42], iv) Image samples of Real-World Masked Face Dataset (RMFD) [54],v) Image samples of Face Mask Dataset [24]

a deep learning one-stage detector (YOLO),(2) a high-performance neural network inference computing framework (NCNN), and (3)a stack-based virtual machine (WebAssembly) [91]. The model has various characteristics such as serverless edge-computing design, compatibility with different devices and OS, free installation, low privacy risk, low response time, etc., and it achieved AP = 89% with high-speed detection. Along with its features, it has some shortcomings like it does not finely decouple the "no mask" and "incorrectly wearing mask" categories; it does not precisely notify the particular incorrect location while a person is incorrectly wearing a mask, such as disclosing the nose or mouth and also, This model does not support parallel computing features.

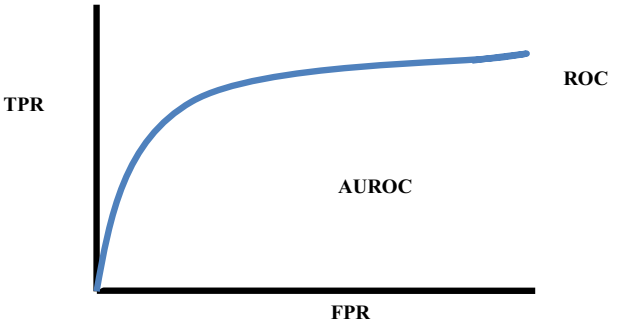



**Fig. 23** AUROC curve plot using TPR and FPR for a classifier

Jiang et al. [24] developed a one-stage face mask detector named "RetinaFaceMask" as it follows the same architecture of RetinaNet. The network architecture of their proposed framework consists of "ResNet" as a backbone network which is used for feature extraction purposes to generate the feature maps and simply built from CNN, and they have also used another pre-trained model, "MobileNet" for the comparison, then FPN is employed as a neck which lies between backbone network and heads. Necks are used for the enhancement of Feature maps. Context attention modules are applied as the heads(predictors or detectors)to boost the detection performance. The FaceMask Dataset they have used is very small-sized, and deep learning models generally require a large amount of training data, so due to this issue, they have applied the concept of Transfer Learning in their model. They have achieved Precision = 93.4% for mask detection by applying RetinaFaceMask and ResNet model.

P Nagrath et al. [59] proposed a real-time framework for the detection of face masks called**"SSDMNV2**," which is composed of a "Single Shot Multibox Detector" that acts as a face mask detector and a classification architecture: "MobilenetV2". The SSDMNV2 model performs pre-processing and Training steps on the whole dataset, then face mask detection is performed using the trained model. In this, the Data augmentation technique is applied to improve the accuracy. Data was then divided into training and testing data then MobilenetV2 was implemented. After the model is trained, the model is utilized for the classification stage (Detection on images and real-time video).SSD would result in a bounding box on the input image if the masked faces were detected by it. They compared their model's performance with various existing models such as LeNet – 5(accuracy = 84.6%) and AlexNet(89.2% = accuracy), and it outperformed all of them by achieving an F1 score of 0.93and an accuracy of 92.64%.

## II. DATASETS USED IN EXISTING FACE MASK DETECTION LITERATU:

**LFW(Labeled Faces in the Wild)** [39]**:** Nieto-Rodr'ıguez et al. [61] has used this dataset in the "training phase," which was proposed by haung; it contains 13,233 face images of celebrities that are collected from the web. These images are in JPEG format having 250 by 250 pixels. The dataset contains 5749 different people, where 1680 people have two or more images, and the remaining 4069 people have just a single image in the dataset [61].In this dataset, each image is labeled with the name of the person present in that specific image which can be used in Face recognition to identify a particular person. They have used the **BAO dataset** [31]for the "testing phase".

**MASKED FACE dataset:** Wei et al. [9] proposed a new dataset called *"MASKED FACE dataset"*. This Dataset is comprised of just 200 images that are collected from the Web. They labeled these images and split the dataset into training and testing sets having 160 and 40 images, respectively. Due to the significantly less number of images in this dataset and deep learning needs a large amount of data for its training purpose to achieve better accuracy, so they chose the "WIDER FACE dataset" as their pre-training dataset.

**ORL face dataset**: Ejaz et al. [21]has used the ORL face dataset. There are ten different images of 40 people in this dataset. Every image size is 92 × 112, with 8-bit grey levels, and these images are in PGM format. They used their own captured images in addition to ORL face images to form a dataset having masked images. In their experiment analysis, they have 500 images in which 300 images were used for the training phase and 200 images were used for the testing phase.

**Moxa3K Benchmark Dataset**: Roy et al. [74] itself created a dataset named" Moxa3K".It contains 3000 images, from which 2800 images and 200 images are in training testing sets, respectively. Out of 3000 images, 678 images are fetched from the Kaggle dataset, 757 images have close-ups of faces, including frontal and side profiles, and 1565 images are obtained from the internet. These images primarily contain the person wearing masks that depicts the ongoing COVID-19 crisis. The images are in JPEG Format. This dataset also contains annotation XML files in YOLO and PASCAL VOC format. To increase the robustness of the detector, the dataset also contains images from crowded areas, blurred images, images with various illumination conditions, having different weather conditions.

**Medical Masks Dataset (MMD) and Larxel's Face Mask Detection Dataset (FMDD):** Loey et al. [53] performed its experiments on two publically available datasets that are MMD and FMD.MMD contains 682 images having 3 k faces wearing medical masks. In contrast, FMDD contains 853 images in PNG format that belong to three different categories: Mask, no mask, incorrectly wearing a mask, and This dataset also consists of corresponding annotation XML files to 853 images.

**MAFA: A Dataset of Masked Faces** [32]**:** Zhang et al. [99] have made their customized dataset which consists of 4672 images in which 4188 images are collected from one of the popular datasets called MAFA, and the remaining 484 images are collected from the internet. **"MAFA"** is a dataset of masked faces which contains 30,811 images and 35,806 masked faces. This dataset has diverse images with various occlusion degrees (Weak, Medium, and Heavy), and types of masks (Simple, Complex Mask, Human Body, Hybrid mask). The author divides the MAFA images into five categories: clean face, hand-masked face, non-hand-masked face, masked incorrect face, and masked correct face. They divide the total images into 3504 and 1168 images for training and testing, respectively, and For Data annotation author has used a labeling tool (labelImg).

**Simulated Masked Face Dataset (SMFD):** G. Jignesh et al. [42] has conducted its experiments on **SMFD** by using their proposed model. This dataset contains 1570 images,785 for simulated masked facial images, and 785 for unmasked facial images. For the training phase, they took 1099 images from both the masked and unmasked classes of the dataset, and the rest 470 images are used for the testing phase.

**RMFD, SMFD, and LFW:** Loey et al. [54] have used three datasets in their proposed model, such as RMFD (Real-World Masked Face Dataset), which is one of an enormous real-world face masked datasets having 5000 images of 525 people

with masks and 90,000 images of same 525 people without masks. Due to the imbalanced nature of this dataset, the author has used 5000 images for faces with masks and without masks with a total of 10,000 images. In their presented work, they used RMFD, and SMFD(1570 images)in training(70% data), validation(10% data), and testing(20% data) phases, whereas the **LFW** dataset(13 k Images) was used only for the testing phase.

**MaskedFace-Net** [10]**, Flickr-Faces-HQ Dataset(FFHQ)** [43] **and CelebA dataset** [51]**:**Yixin Hu et al. [38] used a dataset that contains 5829 images for the face mask detection module(Training phase:4663 images and testing phase: 1166 images) in which 1903 images and 1926 images are collected from two categories of the "MaskedFace-Net" dataset which is a large dataset that consists of total 137,016 images having 67,193 images, and 69,823 images are correctly and incorrectly wearing masks categories, whereas 2000 non-occluded images are selected from FFHQ publicly available dataset that consists of total 70,000 face images of high quality and the images are in PNG file format having 1024 × 1024 resolution. For the overlay module, During the training phase, the author has selected 10,000 images from the **CelebA dataset**, and for the testing phase, they collected images from the celecA and MaskedFace-Net datasets.

**WIDER FACE dataset** [95] **and MAFA** [32]**:** Zekun et al. [91], for the training of their WearMask model, he has performed their experimentation on 9097 images with 17,532 labeled boxes in which 3894 images are collected from the WIDER FACE dataset, 4065 images from MAFA, and the remaining 1138 images are collected from the web. They split the total data into 80% training and validation and 20% testing. WIDER FACE dataset initially consists of 32,203 images with 393, 703 labeled face boxes, whereas MAFA contains 30,811 images and 35,806 masked faces.

**Face Mask Dataset(FMD)** [13]**:** Jiang et al. [24] have used this dataset which consists of 7959 images along with proper annotations (with a mask or without a mask). This dataset is made up of using two datasets: Wider Face [95], having a variety of poses, occlusion, etc., in images, and MAFA [32], having confused images in which faces are masked by hands or other objects instead of physical masks which in turn brings diversity in FMD.FMD contains different types of images in it, such as facial images with or without masks, masked and unmasked faces in one image, incorrectly worn or images having faces covered with objects, etc.

**Real-Time-Medical-Mask-Detection Dataset** [69]: P Nagrath et al. [59] has made their own dataset (5521 images: "with_mask," 5521 images: "without_mask") with the combination of different datasets such as the "Medical Mask Dataset" by Mikolaj Witkowski, which contains 678 images and Prajna Bhandary dataset from PyImageSearch having 1376 images(690 masked facial images and 986 unmasked facial images). The author has divided their dataset into 80% and 20% for training and testing phases, respectively.

## III. PERFORMANCE EVALUATION CRITERIA FOR FACE MASK DETECTION:

According to the nature of the dataset (Balanced, Imbalanced), selection of an appropriate metric is a must, such as "**accuracy**" is considered as a good measure when the dataset is balanced having mask images: non-mask images = 7:3.

Different evaluation criteria are used by different authors to measure the performance of Face Mask Detection algorithms that are described as below:

1) **Confusion Matrix:** It is a Two-dimensional matrix, as shown in Eq. 1. It has two dimensions: "Actual Class" and "Predicted Class". It is one of the performance metrics that is used in a classification Problem (Face Mask Detection) where "Mask" and "No Mask" represent the class labels.

| | | **Actual Class** | |
|---|---|---|---|
| | | Mask | No Mask |
| **Predicted Class** | Mask | TP | FP |
| | No Mask | FN | TN |

- **Terms associated with the confusion matrix are explained as follows:**

  a. True Positive(TP): It is the case when both the predicted class and actual class of an input image is "Mask," it means that the model predicted there is a mask on the face in an image and also in actual, there is a mask existing in an image. It is generally considered as the *best case* of the model.
  b. True Negative(TN): It is the case when both the predicted class and actual class of an input image is "No Mask," which means the model predicts there is no mask present in an input image, and in actual it is True.
  c. False Positive(FP): It is the case when the predicted class is "No mask," and the actual class is "Mask," that is, the model falsely predicts there is no mask in an image, but in actuality, there is a mask present on the face in an image. It is generally considered as the *worst case* of the model. These are also known as ***"Type-1 errors***".
  d. False Negative(FN): It is the case when the predicted class is "Mask," and the actual class is "No Mask," which is also the wrong prediction by the model because actually there is no mask present in an image but model depicts that there is a mask present in an image. These are also known as ***"Type-2 errors***".

2) **Classification Report**: This report includes the following mentioned scores.

  a. Precision: As shown in Eq. 2. Precision is a ratio of **TP** and submission of **TP** and **FP**, which describes what proportion of people wearing a mask predicted by the model are actually wearing a mask. It simply tells us that out of the total predicted positive results, how many results are actually positive.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

b. Recall or Sensitivity: As shown in Eq.3. The recall is a ratio of **TP**(Faces predicted by the model wearing a mask) and submission of **TP** and **FN**(total actual positives that describes a total number of people who are actually wearing a mask); **FN** value is considered here because the Person is actually wearing a mask even if the model predicted its opposite. It is also known as ***the "True Positive Rate***".

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

c. F1 Score: Both Precision and Recall are collectively represented by the F1 score. As shown in Eq.4. it is simply a Harmonic Mean(**2xy/x + y**) of the Precision and Recall.

$$\text{F1 Score} = 2 \times \left( \frac{\text{Precision*Recall}}{\text{Precision} + \text{Recall}} \right) \tag{4}$$

d. Support: It refers to the actual number of samples of the class "Mask" and "No Mask" in the dataset.

3) **Accuracy**: As shown in Eq.5. Accuracy is defined as a ratio of the number of correct predictions made by the Face Mask detection model and the number of total predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{5}$$

4) **Specificity**: This measure depicts that out of the total people who are actually not wearing a face mask (TN + FP) are predicted under the category **"No Mask"** (TN) by the model. **FP** value is considered here because the Person is actually not wearing a mask even if the model predicted that person is wearing a mask. It is also known as the ***"True negative Rate***".

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{6}$$

5) **Confidence score**: It is the probabilistic measure that tells how confident the model is about the prediction it made being correct. Generally, Face mask detectors give bounding boxes along with confidence scores as an output where the confidence score describes how confident our detector is about the presence of an object (Mask on face) in the bounding box.

6) **IOU**: IOU is calculated by dividing the overlapping area of the predicted($B_p$) and ground truth ($B_g$) bounding boxes by the area of union between these two bounding boxes, as shown in Eq.7. Using the IOU and Threshold, we can determine whether a detection made by the face mask detection model is True Positive or False Positive.

**True Positive (TP)** denotes if detection with IOU $\geq$ *threshold,* then it is a correct detection.

**False Positive (FP)** denotes if detection with IOU $<$ *threshold*, then it is a false detection.

$$\text{IOU} = \frac{\text{Area}(\text{Bg} \cap \text{Bp})}{\text{Area}(\text{Bg} \cup \text{Bp})} = \frac{\textit{Area of Intersection}}{\textit{Area of union}} \tag{7}$$

7) **AUROC**: This evaluation metric is used to determine the performance of binary classifiers. **AUC** represents the degree of separability of classes, where the Higher the AUC, the better our model is at differentiating between people wearing a mask or not.ROC is a probability curve having FPR (**False positive rate** = 1-Specificity) on the x-axis and TPR (**True positive rate/sensitivity)** on the y-axis.

## 3 Applications

Due to the increasing need for Face masks to control the Covid-19, a Face mask detection system can be deployed in a real-world environment to check whether the person is wearing a mask or not. Following are some of the crowded public places where wearing a mask is one of the mandatory Covid-19 norms, and to monitor the violation of this norm, the installation of an efficient face mask detection system is a must.

1. **Hospitals:** This is a crucial area where even before Covid-19, doctors preferred to wear a mask during any surgery to prevent the passage of any infections, but nowadays, to reduce the infection rate of Coronavirus, surveillance cameras must be placed in hospitals for the detection of face masks to protect the patients as well as health workers.
2. **Examination halls:** At present, Many Students may wait in queues for their formalities before entering the examination hall, so due avoid overcrowding, face mask detection can be installed for their safety.
3. **Airports/ Stations:** Airports and stations are one of the most restricted places where not wearing a face mask is strictly prohibited. The face mask system can be implemented there to ensure that the travelers and workers are following all the necessary safety protocols for Covid-19.
4. **Shopping malls:** Shopping malls, supermarkets, and cinema halls are some of the crowded places where face mask detection systems can be utilized to safeguard the customers and employees.
5. **Workplaces:** After Work from home, various organizations and educational institutions are now re-opening and making it compulsory for all to wear a mask; therefore, a face mask detection system is required instead of manually checking whether each person is wearing a face mask or not.
6. **Social Gatherings:** Wearing a Mask becomes mandatory in social gatherings such as protests, weddings, rallies, etc., so a face mask detection system should be used there to reduce the disease.

## 4 Challenges and future scope

A. **Facial Masks with Complex color patterns:** Nowadays**,** People are using different types of stylish face masks. Some masks have lips, nose, or chin on it, and some with complex patterns with a variety of colors, So most of the existing works that have been done are mainly oriented toward finding surgical masks, and for a face mask detection system, it is challenging to recognize a person with their personalized masks. It may confuse the system between the natural face and mask and increase the false positives,

reducing accuracy. In the future, to fill this gap, this challenge must be considered to make an efficient system.

B. **Masked Face Recognition:** When faces are covered with masks, the visibility of crucial parts of the face, such as the mouth, nose, chin, etc., has been lowered, due to which face recognition is hampered. It is hard for a system to detect the person's identity when he/she is wearing a mask which may result in criminal offenses, terrorist attacks, etc. So, masked face recognition is a challenging issue that should be considered.

C. **Limited Datasets:** Deep learning-based neural network approaches have been widely used due to their detection rate, but these methods require a large amount of training data to make a robust detector. So, Researchers are nowadays using their own customized datasets due to the problem of the unavailability of large datasets. Some datasets USED in the existing literature have a significantly less number of images that automatically degrade the performance of the detector. The need to balance the dataset should be eliminated in the future by having the same number of masked and unmasked images in it. So in the future, researchers can come up with adequate datasets which will improve the overall performance of the face mask detector.

D. **Hyperparameter Optimization:** There are different hyperparameters, such as Epochs, Learning rate, batch size, etc., that need to be optimized in order to get better results through performance metrics. The best selection of hyperparameters by the researchers helps in controlling the performance of the learning algorithm of the model.

E. **Two-class detection Problem:** Most of the recent researches are mainly focused on the classification of two classes,"with_mask" or "without_mask," which neglects the vital subject of whether the mask is worn correctly or not because the improper wearing of a mask is precisely equal to not wearing a mask at all, as nose or mouth should be adequately covered to decrease this infectious disease rate, in future works, "mask_incorrect" new class should also be considered as well and face mask detector should be able to distinguish between mask worn correctly or incorrectly.

F. **Real-time face mask detection:** To deploy the face mask detection system in the real-world environment, the system should be capable of detecting the masked and unmasked faces from CCTV cameras or live video streams, and execution of this is quite challenging in real-world scenarios. A real-time face mask detector should perform well in all conditions such as different types of masks, occlusions, orientations, weather conditions, etc. There may be a speed-accuracy tradeoff in such types of detectors where an increase in speed can significantly affect the accuracy. Different types of issues can occur, such as memory issues, cameras with low resolution, and a significant distance between the camera and faces resulting in degradation of quality and detection rate.

G. These issues can be resolved in the future by embedding high-resolution equipment, having good computing resources, or training the model in different conditions with adequate datasets.

H. **Impact of Diversity in Data:** A robust face mask detector can be able to detect masked or unmasked frontal facial images, but people are not looking at the CCTV cameras as in reality, they are the moving objects. In real-world environments, there may be different illumination conditions, face orientations, blurry images, different weather conditions, non-mask occlusions, etc., so to improve the performance of the face mask detector, the diversity of the dataset should be increased by training the detector with images of scenarios as mentioned above. It may still be difficult for a detector to detect small facial images.

## 5 Conclusion

This review paper makes a detailed and systematic review of the existing Face mask detection algorithms based on deep learning with reference to Covid-19. This study not only discusses the different object detection algorithms: non-neural algorithms, and Neural algorithms, in detail but also reviews the in-depth exploration of the existing studies performed related to face mask detection. Furthermore, the different datasets USED in these studies, their different experimental evaluation criteria, and the results obtained are summarized. There are different application areas and challenges that are highlighted. This review paper is meaningful for the researchers working in this field by providing conceptual knowledge about existing face mask algorithms, and they can make improvements to those algorithms or develop their novel algorithms to build a powerful facemask detection system. Although remarkable studies have been done to make an efficient face mask detection system from the last two years, there is still room for more improvements and future developments. In the upcoming future, different algorithms can be applied to a widely used dataset, and comparisons between them can be made using various performance metrics.

**Abbreviations** *AdaBoost,* Adaptive Boosting; *AI,* Artificial Intelligence; *AUROC,* Area under the receiver operating characteristic; *BB,* bounding box; *CNN or ConvNet,* Convolutional neural network; *DL,* Deep Learning; *DCNN,* Deep Convolutional Neural Network; *FPN,* Feature pyramid networks; *FPR,* False Positive Rate; *GAN,* Generative Adversarial Network; *HOG,* Histograms of oriented gradients; *HSV,* Hue, Saturation, Value; *IoU,* Intersection over Union; *mAP,* mean average precision; *MERS-CoV,* Middle East Respiratory Syndrome; *ML,* Machine Learning; *NMS,* Non-maximum suppression; *PCA,* Principal Component Analysis; *RCNN,* Region-based Convolutional neural network; *RFCN,* Region-based Fully Convolutional Network; *RoI,* Region of Interest; *RPN,* Region Proposal Network; *SARS-CoV,* Severe Acute Respiratory Syndrome; *SGDM,* Stochastic Gradient Descent with momentum; *SIFT,* Scale-invariant feature transform; *SPPNet,* Spatial Pyramid Pooling Networks; *SURF,* Speeded Up Robust Features; *SSD,* Single Shot Detector; *TL,* Transfer Learning; *TPR,* True positive rate; *WHO,* World Health Organization; *YOLO,* You Look Only Once

**Author contributions** All authors contributed to the study conception and design, material preparation and analysis. All authors read and approved the final manuscript.

**Funding** The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

**Data availability** Not Applicable.

**Code Availability** Not Applicable.

**Declarations** No animal research was conducted by any of the authors in this manuscript.

**Competing interests** The authors have no conflicts of interest to declare that are relevant to the content of this article.

## References

1. "Haar wavelet." (2021) https://en.wikipedia.org/wiki/Haar_wavelet (accessed Sep. 09, 2021)

2. "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way." (2021) https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (accessed Oct. 02, 2021)
3. Adithya K, Babu J (2020) A review on face mask detection using convolutional neural network, pp 1302–1304
4. Advice for the public: Coronavirus disease (COVID-19) (2021) https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public. Accessed 07 Sept 2021
5. Aziz L, Salam MSBH, Sheikh UU, Ayub S (2020) Exploring deep learning-based architecture, strategies, applications and current trends in generic object detection: a comprehensive review. IEEE Access 8: 170461–170495. https://doi.org/10.1109/ACCESS.2020.3021508
6. Bay H, Tuytelaars T, Van Gool L (2006) LNCS 3951 - SURF: Speeded Up Robust Features. Comput Vision–ECCV 2006:404–417. https://doi.org/10.1007/11744023_32
7. Bochkovskiy A, Wang C-Y, Liao H-YM (2020) "YOLOv4: Optimal Speed and Accuracy of Object Detection," 2020, [Online]. Available: http://arxiv.org/abs/2004.10934
8. Bourdev L, Brandt J (2005) "Robust object detection via soft cascade," Proc. - 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, CVPR 2005, vol. 2, pp 236–243, https://doi.org/10.1109/CVPR.2005.310
9. Bu W, Xiao J, Zhou C, Yang M, Peng C (2017) "A cascade framework for masked face detection," 2017 IEEE Transactions on Systems, Man, and Cybernetics: SystemsCIS 2017 IEEE Conference on Robotics and Automation Mechatronics, RAM 2017-Proceedings, vol. 2018-January, pp. 458–462, https://doi.org/10.1109/ICCIS.2017.8274819
10. Cabani A, Hammoudi K, Benhabiles H, Melkemi M (2021) MaskedFace-net – a dataset of correctly/incorrectly masked face images in the context of COVID-19. Smart Health 19:100144. https://doi.org/10.1016/j.smhl.2020.100144
11. Cai Z, Vasconcelos N (2018) Cascade R-CNN: delving into high quality object detection. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit:6154–6162. https://doi.org/10.1109/CVPR.2018.00644
12. Chen K et al (2019) "Hybrid task cascade for instance segmentation," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2019-June, pp 4969–4978, https://doi.org/10.1109/CVPR.2019.00511.
13. Chiang D (2020) Detect faces and determine whether people are wearing mask. https://github.com/AIZOOTech/FaceMaskDetection
14. "Convolutional neural network." (2021) https://en.wikipedia.org/wiki/Convolutional_neural_network (accessed Oct. 02, 2021)
15. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297. https://doi.org/10.1007/BF00994018
16. COVID-19 Action Guide (2021) https://www.ibm.com/thought-leadership/institute-business-value/report/covid-19-action-guide?lnk=hpmex_buco_inen&lnk2=learn. Accessed 08 Sept 2021
17. COVID-19 pandemic (2021) https://en.wikipedia.org/wiki/COVID-19_pandemic#Cause. Accessed 04 Sept 2021
18. Dai J, Li Y, He K, Sun J (2016) "R-FCN: object detection via region-based fully convolutional networks," Adv Neural Inf Process Syst 379–387
19. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. Proc. - 2005 IEEE Comput. Soc. Conf. Comput. Vis Pattern Recognition, CVPR 1:886–893. https://doi.org/10.1109/CVPR.2005.177
20. Deng J, Xuan X, Wang W, Li Z, Yao H, Wang Z (2020) A review of object detection based on deep learning. Multimed Tools Appl 1684:1. https://doi.org/10.1088/1742-6596/1684/1/012028
21. Ejaz MS, Islam MR, Sifatullah M, Sarker A (2019) "Implementation of Principal Component Analysis on Masked and Non-masked Face Recognition,"1st International Conference on Advances in Science, Engineering and Robotics Technology, (ICASERT 2019), no. May 2020, https://doi.org/10.1109/ICASERT.2019.8934543
22. Erhan D, Szegedy C, Toshev A, Anguelov D (2014) "Scalable object detection using deep neural networks," Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit pp. 2155–2162, https://doi.org/10.1109/CVPR.2014.276
23. Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A (2010) The pascal visual object classes (VOC) challenge. Int J Comput Vis 88(2):303–338. https://doi.org/10.1007/s11263-009-0275-4
24. Fan X, Jiang M (2021) "RetinaFaceMask: a single stage face mask detector for assisting control of the COVID-19 pandemic," Proceedings Conference, IEEE International Conference on Systems, Man, and Cybernetics pp 832–837. https://doi.org/10.1109/SMC52423.2021.9659271

25. "Feature Engineering for Images: A Valuable Introduction to the HOG Feature Descriptor. " (2021) https://www.analyticsvidhya.com/blog/2019/09/feature-engineering-images-introduction-hog-feature-descriptor/ (accessed Sep. 15, 2021)
26. Felzenszwalb P, McAllester D, Ramanan D (2008) "A discriminatively trained, multiscale, deformable part model," 26th IEEE Conf Comput Vis Pattern Recognition, CVPR, pp. 0–7, https://doi.org/10.1109/CVPR.2008.4587597
27. Felzenszwalb P, Girshick R, McAllester D, Ramanan D (2013) Cascade object detection with deformable part models. Commun ACM 56(9):97–105. https://doi.org/10.1145/2500468.2494532
28. Feng X, Jiang Y, Yang X, Du M, Li X (2019) Computer vision algorithms and hardware implementations: a survey. Integr VLSI J 69(August):309–320. https://doi.org/10.1016/j.vlsi.2019.07.005
29. Forsyth D (2014) Object detection with discriminatively trained part-based models. Computer (Long Beach Calif) 47(2):6–7. https://doi.org/10.1109/MC.2014.42
30. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139. https://doi.org/10.1006/jcss.1997.1504
31. Frischholz R (2021) Bao face database at the face detection homepage. https://facedetection.com/. Accessed 07 Nov 2021).
32. Ge S, Li J, Ye Q, Luo Z, (2017) "Detecting masked faces in the wild with LLE-CNNs," Proceedings 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, vol. 2017-January, pp 426–434, https://doi.org/10.1109/CVPR.2017.53
33. Girshick R (2015) Fast R-CNN. Proc IEEE Int Conf Comput Vis 2015(Inter):1440–1448. https://doi.org/10.1109/ICCV.2015.169
34. Girshick R, Donahue J, Darrell T, Malik J, (2014) "Rich feature hierarchies for accurate object detection and semantic segmentation," Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit., pp. 580–587. https://doi.org/10.1109/CVPR.2014.81.
35. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp 770–778. https://doi.org/10.1109/CVPR.2016.90
36. He K, Gkioxari G, Dollár P, Girshick R (2020) Mask R-CNN. IEEE Trans Pattern Anal Mach Intell 42(2):386–397. https://doi.org/10.1109/TPAMI.2018.2844175
37. Howard AG et al (2017) MobileNets: efficient convolutional neural networks for mobile vision applications, [Online]. Available: http://arxiv.org/abs/1704.04861
38. Hu Y, Li X (2021) "CoverTheFace: face covering monitoring and demonstrating using deep learning and statistical shape analysis,"[Online]. Available: http://arxiv.org/abs/2108.10430
39. Huang GB, Mattar M, Berg T, Labeled EL, Images R, Learned-miller E (2007) "To cite this version : HAL Id : inria-00321923 Labeled Faces in the Wild : A Database for Studying Face Recognition in Unconstrained Environments," Work. faces in'Real-Life'Images Detect. alignment, Recognit, pp. 7–49
40. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017, vol 2017-Janua, pp 2261–2269. https://doi.org/10.1109/CVPR.2017.243
41. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift, 32nd Int. Conf. Mach. Learn. ICML 2015 vol. 1, pp 448–456
42. Jignesh Chowdary G, Punn NS, Sonbhadra SK, Agarwal S (2020) "Face Mask Detection Using Transfer Learning of InceptionV3," Lecture Notes in Computer Science (LNCS) (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12581 LNCS, pp. 81–90, https://doi.org/10.1007/978-3-030-66665-1_6
43. Karras T, Laine S, Aila T (2021) A style-based generator architecture for generative adversarial networks. IEEE Trans Pattern Anal Mach Intell 43(12):4217–4228. https://doi.org/10.1109/TPAMI.2020.2970919
44. Krizhevsky BA, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. Commun ACM 60(6):84–90
45. Kumaraswamy M, Shaji JS, Sowmya VS, Paul DP (2021) General awareness regarding face masks to combat Covid-19: a comprehensive review. Int J Pharm Sci Rev Res 67(1):24–29. https://doi.org/10.47583/ijpsrr.2021.v67i01.004
46. Law H, Deng J (2018) "CornerNet," Eur. Conf. Comput. Vision(ECCV), pp 765–781
47. Li SZ, Zhang ZQ (2004) FloatBoost learning and statistical face detection. IEEE Trans Pattern Anal Mach Intell 26(9):1112–1123. https://doi.org/10.1109/TPAMI.2004.68
48. Lienhart R, Maydt J (2002) An extended set of Haar-like features for rapid object detection. IEEE Int Conf Image Process 1:900–903. https://doi.org/10.1109/icip.2002.1038171
49. Lin T-Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S (2016) "Feature Pyramid Networks for Object Detection," Dec. 2016, [Online]. Available: http://arxiv.org/abs/1612.03144

50. Lin TY, Goyal P, Girshick R, He K, Dollar P (2017) "Focal Loss for Dense Object Detection," Proceedings of the IEEE International Conference on Computer Vision, vol. 2017-Octob, pp 2999–3007, https://doi.org/10.1109/ICCV.2017.324.
51. Liu Z, Luo P, Wang X, Tang X (2015) "Deep learning face attributes in the wild," Proceedings of the IEEE international conference on computer vision, vol. 2015 Inter, pp. 3730–3738, https://doi.org/10.1109/ICCV.2015.425
52. Liu W et al (2016) "SSD: Single shot multibox detector," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9905 LNCS, pp 21–37, https://doi.org/10.1007/978-3-319-46448-0_2
53. Loey M, Manogaran G, Taha MHN, Khalifa NEM (2021) "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection," Sustainable Cities and Society, vol. 65, (November 2020), p 102600, https://doi.org/10.1016/j.scs.2020.102600
54. Loey M, Manogaran G, Taha MHN, Khalifa NEM (2021) A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. Meas J Int Meas Confed 167(July 2020):108288. https://doi.org/10.1016/j.measurement.2020.108288
55. Lowe DG (1999) Object recognition from local scale-invariant features. Proc IEEE Int Conf Comput Vis 2:1150–1157. https://doi.org/10.1109/iccv.1999.790410
56. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110. https://doi.org/10.1023/B:VISI.0000029664.99615.94
57. Mask use in the context of COVID-19 (2021) https://www.who.int/publications/i/item/advice-on-the-use-of-masks-in-the-community-during-home-care-and-in-healthcare-settings-in-the-context-of-the-novel-coronavirus-(2019-ncov)-outbreak. Accessed 07 Sept 2021
58. Msonda P, Uymaz SA, Karaağaç SS (2020) Spatial pyramid pooling in deep convolutional networks for automatic tuberculosis diagnosis. Trait du Signal 37(6):1075–1084. https://doi.org/10.18280/TS.370620
59. Nagrath P, Jain R, Madan A, Arora R, Kataria P, Hemanth J (2021) SSDMNV2: a real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. Sustain Cities Soc 66(August 2020):102692. https://doi.org/10.1016/j.scs.2020.102692
60. Najibi M, Rastegari M, Davis LS (2016) "G-CNN: An Iterative Grid Based Object Detector," Proc. IEEE Comput Soc Conf Comput Vis Pattern Recognit, vol. 2016-Decem, pp 2369–2377, https://doi.org/10.1109/CVPR.2016.260
61. Nieto-Rodríguez A, Mucientes M, Brea VM (2015) "System for Medical Mask Detection in the Operating Room Through Facial Attributes" pp. 138–145. https://doi.org/10.1007/978-3-319-19390-8_16
62. Nowrin A, Afroz S, Rahman MS, Mahmud I, Cho YZ (2021) Comprehensive review on facemask detection techniques in the context of Covid-19. IEEE Access 9:106839–106864. https://doi.org/10.1109/ACCESS.2021.3100070
63. Object detection (2021) https://en.wikipedia.org/wiki/Object_detection. Accessed 08 Sept
64. Opitz D, Maclin R (1999) Popular Ensemble Methods: An Empirical Study. J Artif Intell Res 11(April):169–198. https://doi.org/10.1613/jair.614
65. Origins of the SARS-CoV-2 virus (2021) https://www.who.int/health-topics/coronavirus/origins-of-the-virus. Accessed 03 Sept 2021
66. Ouyang W, Wang K, Zhu X, Wang X (2017) "Chained Cascade Network for Object Detection," Proc. IEEE Int. Conf. Comput. Vis., vol. 2017-Octob, pp 1956–1964, https://doi.org/10.1109/ICCV.2017.214
67. Phung VH, Rhee EJ (2019) A High-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets. Appl Sci 9:21. https://doi.org/10.3390/app9214500
68. Pitts W, McCulloch WS (1947) How we know universals the perception of auditory and visual forms. Bull Math Biophys 9(3):127–147. https://doi.org/10.1007/BF02478291
69. Real-Time-Medical-Mask-Detection Dataset (n.d.). https://github.com/TheSSJ2612/Real-Time-Medical-Mask-Detection/releases/download/v0.1/Dataset.zip
70. Redmon J, Farhadi A (2017) "YOLO9000: Better, faster, stronger," Proceedings 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, vol. 2017-January, pp 6517–6525, https://doi.org/10.1109/CVPR.2017.690
71. Redmon J and Farhadi A (2018) "YOLOv3: An Incremental Improvement," [Online]. Available: http://arxiv.org/abs/1804.02767
72. Redmon J, Divvala S, Girshick R, Farhadi A (2016) "You only look once: Unified, real-time object detection," Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit, vol. 2016-Decem, pp 779–788, https://doi.org/10.1109/CVPR.2016.91
73. Ren S, He K, Girshick R, Sun J (2017) Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell 39(6):1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031

74. Roy B, Nandy S, Ghosh D, Dutta D, Biswas P, Das T (2020) MOXA: a deep learning based unmanned approach for real-time monitoring of people wearing medical masks. Trans Indian Natl Acad Eng 5(3): 509–518. https://doi.org/10.1007/s41403-020-00157-z
75. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536. https://doi.org/10.1038/323533a0
76. Salve SG, Jondhale KC (2010) Shape matching and object recognition using shape contexts. Proc - 2010 3rd IEEE Int Conf Comput Sci Inf Technol ICCSIT 2010 9(24):471–474. https://doi.org/10.1109/ICCSIT.2010.5565098
77. "Scale-invariant feature transform." (2021) https://en.wikipedia.org/wiki/Scale-invariant_feature_transform (accessed Sep. 22, 2021)
78. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. 3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc., pp 1–14
79. Suard F, Rakotomamonjy A, Bensrhair A, Broggi A (2006) Pedestrian detection using infrared images and histograms of oriented gradients. IEEE Intell Veh Symp Proc:206–212. https://doi.org/10.1109/ivs.2006.1689629
80. Szegedy C, Reed S, Erhan D, Anguelov D, and Ioffe S (2014) "Scalable, High-Quality Object Detection," 2014, [Online]. Available: http://arxiv.org/abs/1412.1441
81. Szegedy C et al (2015) Going deeper with convolutions. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit, vol 07-12-June, pp 1–9. https://doi.org/10.1109/CVPR.2015.7298594
82. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp 2818–2826. https://doi.org/10.1109/CVPR.2016.308
83. Szegedy C, Ioffe S, Vanhoucke V, Alemi AA (2017) Inception-v4, inception-ResNet and the impact of residual connections on learning. 31st AAAI Conf. Artif. Intell. AAAI, pp 4278–4284
84. The Possibility of COVID-19 after Vaccination: Breakthrough Infections (2021) https://www.cdc.gov/coronavirus/2019-ncov/vaccines/effectiveness/why-measure-effectiveness/breakthrough-cases.html. Accessed 07 Sept 2021
85. Uijlings JRR, Van De Sande KEA, Gevers T, Smeulders AWM (2013) Selective search for object recognition. Int J Comput Vis 104(2):154–171. https://doi.org/10.1007/s11263-013-0620-5
86. "Understanding Feature Pyramid Networks for object detection (FPN)." (2021) https://jonathan-hui.medium.com/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c (accessed Oct. 07, 2021)
87. Vibhuti, Jindal N, Singh H, Rana PS (2022) Face mask detection in COVID-19: a strategic review. Multimed Tools Appl. https://doi.org/10.1007/s11042-022-12999-6
88. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 1:1–8. https://doi.org/10.1109/cvpr.2001.990517
89. Viola P, Jones M (2004) Robust real-time face detection. Int J Comput Vis 57(2):137–154
90. "Viola–Jones object detection framework." (2021) https://en.wikipedia.org/wiki/Viola–Jones_object_detection_framework (accessed Sep. 09, 2021)
91. Wang Z, Wang P, Louis PC, Wheless LE, Huo Y (2021) WearMask: Fast In-browser Face Mask Detection with Serverless Edge Computing for COVID-19. no. December, pp 1–8, [Online]. Available: http://arxiv.org/abs/2101.00784
92. WHO Coronavirus (COVID-19) (2021) https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/naming-the-coronavirus-disease-(covid-2019)-and-the-virus-that-causes-it). Accessed 03 Sept 2021
93. WHO Coronavirus (COVID-19) (2021) Dashboard https://covid19.who.int/. Accessed 06 Sept 2021
94. Wu X, Sahoo D, Hoi SCH (2020) Recent advances in deep learning for object detection. Neurocomputing 396:39–64. https://doi.org/10.1016/j.neucom.2020.01.085
95. Yang S., Luo P, Loy CC, Tang X (2016) WIDER FACE: a face detection benchmark. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 2016-Decem, pp 5525–5533. https://doi.org/10.1109/CVPR.2016.596
96. "YOLO-V5" (n.d.) https://github.com/ultralytics/yolov5
97. Yoo D, Park S, Lee JY, Paek AS, Kweon IS (2015) "Attentionnet: Aggregating weak directions for accurate object detection," Proc IEEE Int Conf Comput Vis, vol. 2015 Inter, pp 2659–2667, https://doi.org/10.1109/ICCV.2015.305
98. Zhang S, Wen L, Bian X, Lei Z, Li SZ (2018) "Single-shot refinement neural network for object detection, " Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit, pp. 4203–4212, https://doi.org/10.1109/CVPR.2018.00442

99. Zhang J, Han F, Chun Y, Chen W (2021) A novel detection framework about conditions of wearing face mask for helping control the spread of COVID-19. IEEE Access 9:42975–42984. https://doi.org/10.1109/ACCESS.2021.3066538
100. Zhao ZQ, Zheng P, Xu ST, Wu X (2019) Object detection with deep learning: a review. IEEE Trans Neural Networks Learn Syst 30(11):3212–3232. https://doi.org/10.1109/TNNLS.2018.2876865
101. Zou Z, Shi Z, Guo Y, Ye J (2019) Object detection in 20 years: a survey, pp 1–39 [Online]. Available: http://arxiv.org/abs/1905.05055