YOLOv4: A Face Mask Detection System

Akanksha Soni[1], Avinash Rai[2]

[1]Ph.D. Scholar, Dept. of Electronics and Communication Engineering, UIT-RGPV, Bhopal,462033, India
[2]Asst prof, Dept. of Electronics and Communication Engineering, UIT-RGPV, Bhopal,462033
India

Abstract – *Corona virus disease of 2019 or COVID-19 is a rapidly spreading viral infection that has affected millions all over the world. The greatest risk of transmission exists. In public locations one of the most efficient methods to be careful is to wear a mask. However, some irresponsible people refuse to wear face mask with so many excuses. Moreover, developing the face mask detector is very crucial in this case. In this work, open CV is utilized to locate people who are wearing masks. Using real-time video processing, we will develop a deep learning model that can be used to evaluate the ratio of people wearing masks to those who aren't in crowded places. We evaluate the video stream using a real-time video camera and issue a notification when the zone contains persons who are not wearing masks. We used YOLOv4 to determine whether the mask is worn correctly on the face. Darknet framework is employ for YOLO training, which defines the network's architecture and aids CPU and GPU processing. We utilized Tkinter from the Python GUI for the user interface.*

Key Words: Deep Learning, Face Mask Detection, Object detection, Open CV, Darknet, YOLOv4

1. INTRODUCTION

Even though the majority of people in India have been vaccinated, masks are necessary in populated areas because the majority of people do not use masks and do not practice social distance. In our work, we utilize Yolov4 to recognize faces with and without masks. It employs cspdarknet53 as a backbone for feature extraction, and PANet is employed for feature aggregation, which serves as the algorithm's neck. This project is delivered as software that is extremely user friendly. We utilised the Python GUI library, and the user interface was provided by Tkinter. The interface allows users to give multiple forms of input for processing. We used nvidia for CPU and GPU computation. This gives improved performance by providing GPU utilization, GPU memory access and usage, Power usage and temperatures, Time to solution. They are a major element of today's artificial intelligence infrastructure, and new GPUs have been designed and tuned particularly for deep learning.

You Only Look Once is a method for quickly recognizing objects (YOLO). It is an object identification system that is capable of quickly locating objects in images, real-time coverage, and video streams. Object recognition is one of the most challenging problems in image processing. Although there are other methods of object identification, in this work we will focus on YOLOv4. The advantage of YOLO is that it is faster than other networks while keeping accuracy. When tested, the complete image is examined, enabling the model to draw conclusions about the image's broader context.

What does the COCO record mean in YOLO?

Common Objects in Context (COCO) object detection, instance segmentation, image captioning, and human hotspot localization are some of the areas where COCO is expected to support future studies. COCO is a comprehensive data set for object detection, segmentation, and labelling.

1.1 Motivation

It is difficult for the individual to constantly check on the video at all times. As a result, we developed software that alerts authorities if the number of persons who are not wearing masks exceeds the limit we set. Also, give a user interface via which users may manually evaluate the picture and video. Since we are using Yolov v4 and OpenCV for processing, the accuracy is greater than in earlier models.

1.2 Contributions

In this paper, we offer software that will shorten the time authorities spend on-screen examining the covid transmission area. Because of the employment of the YOLOv4 object detecting algorithm, it outperforms the prior models. The contributions are summarized as follows:

- Designed the deep learning based object recognition system to detect whether a mask is worn or not.
- A survey on the key difficulties in face mask detection, which might be useful for developing new face mask detectors in the future.
- Using the Tkinter module of the Python library to provide a user interface.
- Utilized CSPDarknet53 as the backbone and PANet to aggregate features.

2. RELATED WORK

Chaitali & Wanjale [1] ‘Survey On Image Classification Methods. In Image Processing’ This study provides an overview of different supervised classification algorithms that are utilised in image classification. Non-parametric picture classification is the most popular method. This overview presents a variety of classification methods, each with its own set of restrictions.

Manoj Krishna, Neelima, Harshali, Venu Gopala [2] "Image Classification Using Deep Learning" For testing and validation of picture categorization using deep learning, 3 test photos from the AlexNet database were chosen: a sea anemone, a barometer, & stethoscope. AlexNet architecture use a CNN for classification. The trials reveal that the photographs are accurately identified even for a percentage of the test images, demonstrating the efficiency of the deep learning system.

Mingyuan Xin, Yong Wang [3] "Research on image classification model based on deep convolution neural network" Scaling, translation, and different types of distortion-invariant pictures are recognized using deep convolutional neural networks. To avoid manual feature extraction, the convolutional network employs a feature detection layer to understand from training data passively, and neurons on the same feature mapping surface have the same weight due to the weight sharing mechanism.

It uses YOLO v3 algorithm [4], Open Cv, Deep leaning mechanism which is an object detection model to count total number of people who are not wearing masks by taking live camera feed.

It uses the method of deep learning mechanism [5] this model will check each individual person in the crowd weather they wearing mask or not, If not his picture cropped and sent to higher authority to take action on him.

In Paper [6] YOLO v4 and deep learning method is used to checks different types of masks and moving person through live feed camera.

The paper [7] uses Tensor Flow and Open CV are highly recommended in organization and checks each individual persons wearing mask or not, if not his picture is sent match to database of the organization and a warning message will sent through their Gmail.

In Paper [8] the author uses YOLO and R-CNN for helps to identify different types of masks and helps to find the person is wearing mask or not in real time.

3. SYSTEM OVERVIEW

3.1. Object Detection Based on Deep Learning

There are currently two popular deep learning algorithms of object detection: one-stage object detection and two-stage object detection (fig.1). One is the R-CNN algorithm based on Region Proposal, like R-CNN, Fast R-CNN, and Faster R-CNN etc. They are two-stage and require the first use of heuristic methods for example Selective search, or CNN network to generate Region Proposal and then perform classification and regression on Region Proposal. The other is one-stage algorithms such as Yolo and SSD, which only use a CNN network to directly predict the categories and positions of different targets.

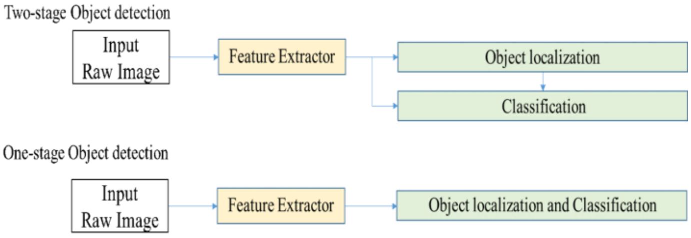

Fig-1: Two stage vs. one stage object detection models

Yolo stands for "You only look once". In the Deep learning era, Yolo is the real-time one stage object detector proposed by Redmon et al. in 2015 [9]. From 2015 onwards, various improvements (different versions V1, V2, V3, V4, V5) were made by the author. The first three versions are researched and developed by the author of the YOLO algorithm, Joseph Redmon. YOLOv4 published by Alexey Bochkovskiy while researcher Glenn Jocher developed the YOLOv5. YOLO is an extremely fast unified, Real-Time Object Detection model, simple to construct and can be trained directly on the full image/video.

3.1.1 Deep learning algorithm – YOLOv4:

We utilized the deep learning approach in YOLOv4 to construct a facial recognition model. Object detection models are all YOLO models. Object detection models are trained to scan an image for a variety of different sorts of objects. These object classes are wrapped in a bounding box and their class is identified when they are detected. Object detection models are typically, object detection models are trained and assessed using the COCO dataset, which has 80 different item types. It is thus anticipated that if object detection models are exposed to additional training data, they will generalize to new object identification tasks. YOLOv4 prioritizes real-time detection and trains on a single GPU. The developers want for vision engineers and developers to be capable of utilizing their YOLOv4 framework in customized domains with ease. This is what we've done here: we've trained the Yolov4 model to distinguish face with masks and without mask. Typically, the backbone network for an object detector is pre-trained on ImageNet classification. The network's weights have previously been tuned to identify key aspects in a picture, however they will be altered in the additional duty of object detection. The CSPDarknet53 are designed to alleviate processing constraints in the DenseNet and enhance learning by transmitting an unedited version of the feature map.

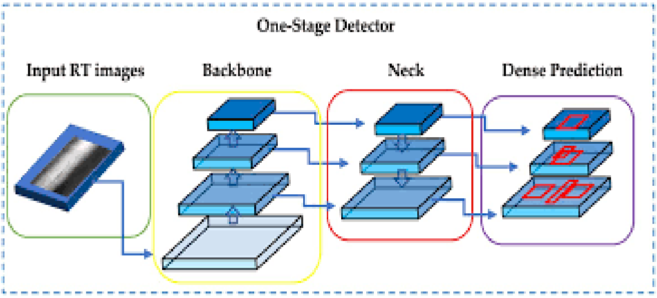

Fig- 2: YOLOv4 architecture [10]

Feature aggregation is the next step following feature extraction. The next step in object detection is to mix and combine the features produced in the ConvNet backbone to prepare for the detection stage. PANet is chosen by YOLOv4 for network feature aggregation. After CSPDarknet53, YOLOv4 contains an SPP block to enlarge the receptive region and separate the most important characteristics from the backbone.

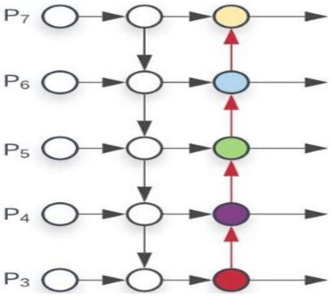

Fig-3: PANet

YOLOv4 employs a "Bag of Freebies," which improve network speed without increasing production inference time. The vast bulk of the freebies in the Bag of Freebies are data augmentation-related. In YOLOv4, we wrote an in-depth look into Data Augmentation, and we'll go over the tactics below . Data augmentation is crucial in computer vision, and we strongly recommend it to get the most out of your models. YOLOv4 implements "Bag of Specials" methods, so named because they add minimal inference time but dramatically improve performance, making them valuable. For GPU processing, we utilized nvidia, which is primarily used to improve the performance and speed of the execution.

3.2. OpenCV

OpenCV is an open-source software library for computer vision and machine learning and for free. OpenCV has been created to offer a standard foundation for computer vision applications and to speed up the incorporation of machine perception into commercial goods. We utilized OpenCV to collect the input, process the input, manage the data flow, get the output from the Yolov4 model, save it in the output folder, and present the output to the user all at the same time.

3.3. Tkinter

Python's interface to the Tk GUI toolkit, which is included with Python, is called Tkinter. This is something we'll investigate further in this chapter. Python's standard GUI library is Tkinter. When Python is used in conjunction with Tkinter, creating graphical user interface (GUI) applications is simple and rapid. Tkinter adds a powerful object-oriented interface to the Tk GUI toolkit. Button, Checkbutton, Entry, Frame, Label, LabelFrame, Menubutton, PanedWindow, Radiobutton, Scale, Scrollbar, and Spinbox are among the widgets available through the tkinter.ttk module. Combobox, Notebook, Progressbar, Separator, Sizegrip, and Treeview are the remaining six widgets, which are all instances of user interface elements.

4. IMPLEMENTATION

4.1 Creating User interface using tkinter

Tkinter is a Python binding tool provided with Microsoft Windows. We use the Tkinter module from Python to create the project's user interface. The file dailog module is used for file uploading section and Message Box Widget is used to display the message boxes in the applications. We also use tkinter.ttk to style our widgets in the same way that HTML styles are styled.

4.2 Processing photo and video through YOLOv4

The input picture or video is analyzed with yolov4, which employs a predetermined parameter to discover faces with and without masks. Similarly, in the case of video, the input file is analyzed as a frame every 5 frames. The input file is handled and managed using OpenCV.

4.3 Processing real-time video through YOLOv4

The user picks a realtime video stream from the User interface. Using OpenCV, we get access to the webcam or any live feed camera, then process each frame of the video, verify the output parameter, and store the result in the output folder, as well as display it to the user

4.4 Enhancing the output parameter

In this phase, we review the output of the analyzed frame or image to see if the proportion of uncovered persons is larger than 20%. If this is the case, the status will be changed to danger and the proper authorities will be notified. When it is 10% to 19%, we just change the status to warning so that no authorities are notified. If it falls below 10%, the classification is changed to safe.

5. CONCLUSIONS

Face mask detection primarily focuses on lowering on-screen time for relevant authorities seeking to prevent covid transmission. We provide a user interface that allows consumers to pick the type of processing they desire. They can analyze a picture, video, or real-time video. Since we started using Yolov4, the performance has been superior to any previous model that came before it. We utilized CSPDarknet53 as the backbone and PANet to aggregate features. The outcome shows the warning status, and the authority has been warned that the region is at high risk of covid transmission. As a result, the authority's burden is lowered, and the covid transmission is minimized and maintained under control.

In the future, the proposed vision system is expected to be employed in various applications. For example this system could be widely used in surveillance systems which is not only limited to the pandemic. As an example wearing a mask could be important due to air pollution. Also robotic applications such as mobile robots could use such system as part of their vision module. It is also expected that through the above improvements, such a system can be applied to outdoor scenes in addition to indoors. Since wearing a mask is an important measure for prevention of spreading the virus, hopefully such system could be beneficial for developing new tools and technologies for the future pandemics.

REFERENCES

[1] Chaitali Dhaware , Mrs . K. H. Wanjale , "Survey On Image Classification Methods In Image Processing" , IJCST June – 2016

[2] Manoj Krishna , Neelima , Harshali , Venu Gopala Rao , "Image Classification using Deep Learning" , IJCSE March – 2018

[3] Mingyuan Xin, Yong Wang, "Research on image classification model based on deep convolution neural network", EURASIP Journal on Image and Video Processing, 2019. DOI:10.1186/s13640-019-0417-8

[4] Prithvi N. Amin; Sayali S. Moghe; Sparsh N. Prabhakar; Charusheela M. Nehete, "Deep Learning Based Face Mask Detection and Crowd Counting", 2021 6th International Conference for Convergence in Technology (I2CT)

[5] Mohammad Marufur Rahman; Md. Motaleb Hossen Manik; Md. Milon Islam;, "An Automated System to Limit COVID-19 Using Facial Mask Detection in Smart City Network", 2020 IEEE International IOT, Electronics Mechatronics Conference (IEMTRONICS)

[6] Susanto Susanto; Febri Alwan Putra; Riska Analia; Ika Karlina Laila Nur Suciningtyas, "The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam", 2020 3rd International Conference on Applied Engineering (ICAE)

[7] Harish Adusumalli; D. Kalyani; R.Krishna Sri; M. Pratapteja, "Face Mask Detection Using OpenCV", 2020 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV).

[8] Jun Zhang; Feiteng Han; Yutong Chun; Wang Cheno," A Novel Detection Framework About Conditions of Wearing Face Mask for Helping Control the Spread of COVID-19", 2020 IEEE Access (Volume: 9)

[9] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in: 2016: pp. 779–788

[10] Zhi-Hao Chen, Jyh-Ching Juang, "YOLOv4 Object Detection Model for Nondestructive Radiographic Testing in Aviation Maintenance Tasks" AIAA JOURNAL, DOI: 10.2514/1.J060860