iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 Issue: VI Month of publication: June 2021

DOI: https://doi.org/10.22214/ijraset.2021.35280

Real Time Face Mask Identification using AI-Deep Learning Neural Network

Aishwarya. B. K[1], Chandan. L[2], Ganesh. B[3], Harshitha. C[4], Sreenivasulu. K. N[5]
[1, 2, 3, 4, 5]*Electronics and Communications Engineering, Nagarjuna College Of Engineering And Technology, Bengaluru*

Abstract: The COVID - 19 pandemic is devastating mankind irrespective of caste, creed, gender, and religion. Contribution of each individual to constrain the expansion of the corona- virus. Is a primary objective/Fundamental duties as a responsible individual to Use a face mask can undoubtedly help in managing the spread of the virus. COVID - 19 face mask Detector uses or owns Facemask net, deep learning techniques to successfully test whether a person is with wearing a face mask or not.

In this project we are working on "FACE MASK IDENTIFICATION USING AI DEEP LEARNING NEURAL NETWORK". The end of 2019 witnessed the outbreak of Corona virus Disease 2019 (COVID-19), which has continued to be the cause of plight for millions of lives and businesses even in 2020. As the world recovers from the pandemic and plans to return to a state of normalcy, there is a wave of anxiety among all individuals, especially those who intend to resume in-person activity. Studies have proved that wearing a face mask significantly reduces the risk of viral transmission as well as provides a sense of protection. However, it is not feasible to manually track the implementation of this policy. Technology holds the key here. We are using a Deep Learning based system that can detect instances where face masks are not used properly. Our system consists of a faster region-based Convolution Neural Network (FRCNN) architecture capable of detecting masked and unmasked faces and can be integrated with preinstalled CCTV cameras. This will help track safety violations, promote the use of face masks, and ensure a safe working environment.

I. INTRODUCTION

Object recognition is to describe a collection of related computer vision tasks that involve activities like identifying objects in digital photographs. Image classification involves activities such as predicting the class of one object in an image. Object localization is referring to identifying the location of one or more objects in an image and drawing an abounding box around their extent. Object detection does the work of combines these two tasks and localizes and classifies one or more objects in an image.

One of the further extensions to this breakdown of computer vision tasks is object segmentation, also called "object instance segmentation" or "semantic segmentation," where instances of recognized objects are indicated by highlighting the specific pixels of the object instead of a coarse bounding box. From this breakdown, we can understand that object recognition refers to a suite of challenging computer vision tasks. For example, image classification is simply straight forward, but the differences between object localization and object detection can be confusing, especially when all three tasks may be just as equally referred to as object recognition. Humans can detect and identify objects present in an image. The human visual system is fast and accurate and can also perform complex tasks like identifying multiple objects and detect obstacles with little conscious thought. The availability of large sets of data, faster GPUs, and better algorithms, we can now easily train computers to detect and classify multiple objects within an image with high accuracy. We need to understand terms such as object detection, object localization, loss function for object detection and localization. Image classification also involves assigning a class label to an image, whereas object localization involves drawing a bounding box around one or more objects in an image. Object recognition refers to a collection of related tasks for identifying objects in digital photographs. Region-based Convolutional Neural Networks, or R-CNNs, is a family of techniques for addressing object localization and recognition tasks, designed for model performance.

II. LITERATURE SURVEY

A. The problem of detecting objects in still, Gray-scale images. Our primary focus is the development of a learning-based approach to the problem that makes use of a sparse, part-based representation. A vocabulary of distinctive object parts is automatically constructed from a set of sample images of the object class of interest; images are then represented using parts from this vocabulary, together with spatial relations observed among the parts. Based on this representation, a learning algorithm is used to automatically learn to detect instances of the object class in new image.

B. The present generic object measure, quantifying how likely it is for an image window to contain an object of any class. We explicitly train it to distinguish objects with a well- defined boundary in space, such as cows and telephones, from amorphous

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

background elements, such as grass and road. The measure combines in a Bayesian framework several image cues measuring characteristics of objects, such as appearing different from their surroundings and having a closed boundary. This includes an innovative cue measuring the closed boundary Characteristic**.**

C. An observer is called active when engaged in some kind of activity whose purpose is to control the geometric parameters of the sensory apparatus. The purpose of the activity is to manipulate the constraints underlying the observed phenomena in order to improve the quality of the perceptual results. For example, a monocular observer that moves with a known or unknown motion or a binocular observer that can rotate his eyes and track environmental objects are just two examples of an observer that we call active. We prove that an active observer can solve basic vision problems in a much more efficient way than a passive one**.**

D. Object recognition systems constitute a deeply entrenched and omnipresent component of modern intelligent systems. Research on object recognition algorithms has led to advances in factory and office automation through the creation of optical character recognition systems, assembly-line industrial inspection systems, as well as chip defect identification systems. It has also led to significant advances in medical imaging, defenses and biometrics. In this paper we discuss the evolution of computer-based object recognition systems over the last fifty years, and overview the successes and failures of proposed solutions to the problem**.**

E. Object recognition systems constitute a deeply entrenched and omnipresent component of modern intelligent systems. Research on object recognition algorithms has led to advances in factory and office automation through the creation of optical character recognition systems, assembly-line industrial inspection systems, as well as chip defect identification systems. It has also led to significant advances in medical imaging, defenses and biometrics. In this paper we discuss the evolution of computer-based object recognition systems over the last fifty years, and overview the successes and failures of proposed solutions to the problem. We survey the breadth of approaches adopted over the years in attempting to solve the problem, and highlight the important role that active and attentive approaches must play in any solution that bridges the semantic gap in the proposed object representations, while simultaneously leading to efficient learning and inference algorithms. From the earliest systems which dealt with the character recognition problem, to modern visually-guided agents that can purposively search entire rooms for objects, we argue that a common thread of all such systems is their fragility and their inability to generalize as well as the human visual system can.

F. The remarkable abilities of the primate visual system have inspired the construction of computational models of some visual neurons. We propose trainable hierarchical object recognition model, which we call S-COSFIRE (S stands for Shape and COSFIRE stands for Combination of Shifted Filter Responses) and use it to localize and recognize objects of interests embedded in complex scenes. It is inspired by the visual processing in the ventral stream (V1/V2 → V4 → TEO). Recognition and localization of objects embedded in complex scenes is important for many computer vision applications. Most existing methods require prior segmentation of the objects from the background which on its turn requires recognition [6]. [

G. Pedestrian detection continues to hold a significant role in the concept, analysis and function of computer vision. Deep learning techniques in pedestrian detection have demonstrated powerful results in recent experiments and research. In this paper a powerful deep learning technique of R-CNN is evaluated for Pedestrian detection on two different pedestrian detection datasets. The experiment involves the use of a deep learning feature extraction model along with the R-CNN detector. The deep learning feature extraction used is the Alexnet. Transfer learning is performed on the feature extraction model to adjust the weights of the convolutional neural networks to favour classification on the selected datasets. The R-CNN detector is then trained on the deep learning feature extraction model for pedestrian detection. The results of the experiments as evidently demonstrated, indicate some important truths about the performance of R-CNN detector on varying datasets.

H. State-of-the-art object detection networks depend on region proposal algorithms to hypothesize object locations. Advances like SPPnet [1] and Fast R-CNN [2] have reduced the running time of these detection networks, exposing region proposal computation as a bottleneck. In this work, we introduce a Region Proposal Network (RPN) that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. An RPN is a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position. The RPN is trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection. We further merge RPN and Fast R-CNN into a single network by sharing their convolutional features-using the recently popular terminology of neural networks with 'attention' mechanisms, the RPN component tells the unified network where to look. For the very deep VGG-16 model [3], our detection system has a frame rate of 5 fps (including all steps) on a GPU, while achieving state-of-the-art object detection accuracy on PASCAL VOC 2007, 2012, and MS COCO datasets with only 300 proposals per image. In ILSVRC and COCO 2015 competitions, Faster R-CNN and RPN are the foundations of the 1st-place winning entries in several tracks.

I. Deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, they achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way SoftMax. To make training faster, they used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout".

J. Even Though tremendous strides have been made in uncontrolled face detection, accurate and efficient face localization in the wild remains an open challenge. This paper presents a robust single-stage face detector, named Retina Face, which performs pixel-wise face localization on various scales of faces by taking advantages of joint extra-supervised and self- supervised multi-task learning.

III. METHODOLOGY

1) Inception-V3: All paragraphs must be indented. All paragraphs must be justified, i.e., both left-justified and right-justified.
Inception-v3 is widely used as image recognition model that has showed to obtain accuracy of greater than 78.1% on the ImageNet dataset. The model is the culmination of many ideas developed by researchers over years. It is based on "Rethinking the Inception Architecture Computer Vision" by Szegedy. The model is made of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concats, dropouts, and fully connected layers. Batch norm is used more throughout the model and applied to activation inputs. Loss is computed via SoftMax.

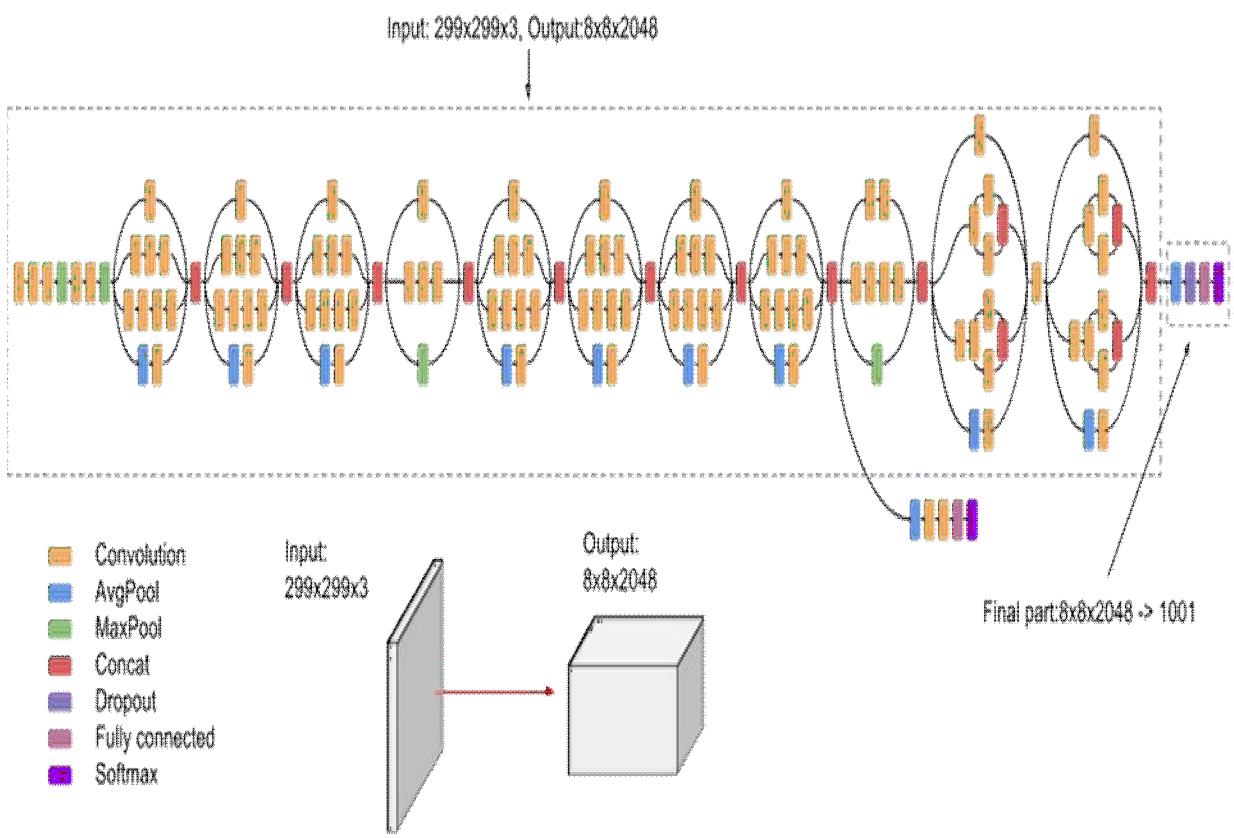

Fig-1: A high-level diagram of the model of Inception-V3

Inception-v3 is a convolutional neural network architecture from the Inception family that makes several improvements including using Label Smoothing, factorized 7 x 7 convolutions, and the use of an auxiliary classifier to propagate label information lower down the network (along with the use of batch normalization for layers in the side head).

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

2) *Faster R-CNN:* Both of the algorithms (R-CNN & Fast R-CNN) uses selective search to find out the region proposals. Selective search is the slow and time-consuming process which affect the performance of the network. Similar to Fast R-CNN, the image is provided as an input to a convolution network which provides a convolutional feature map. Instead of using the selective search algorithm for the feature map to identify the region proposals, a separate network is used to predict the region proposals. The predicted the region which is proposals are then reshaped using an RoI pooling layer which is used to classify the image within the proposed region and predict the offset values for the bounding boxes.

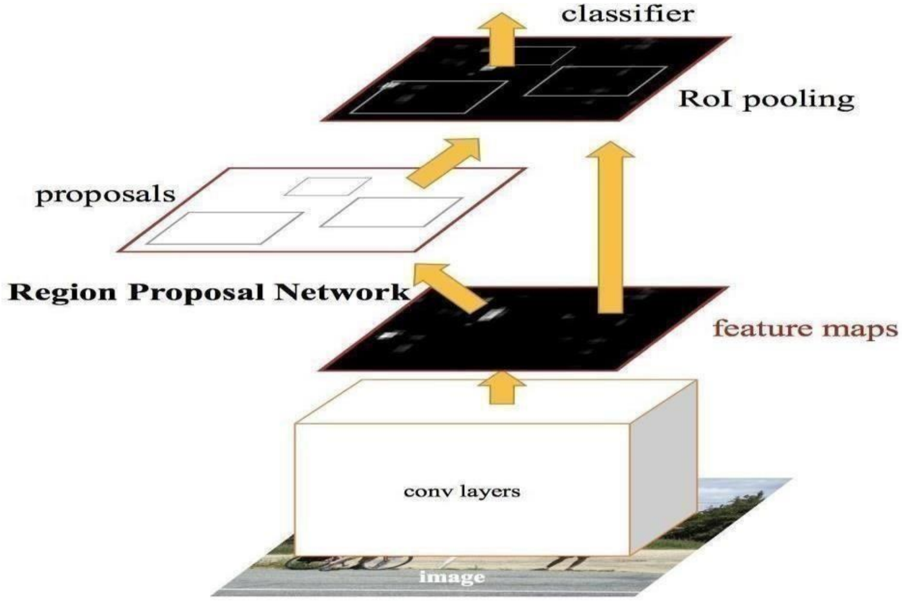

Fig-2: FRCNN model

3) *Block Diagram*

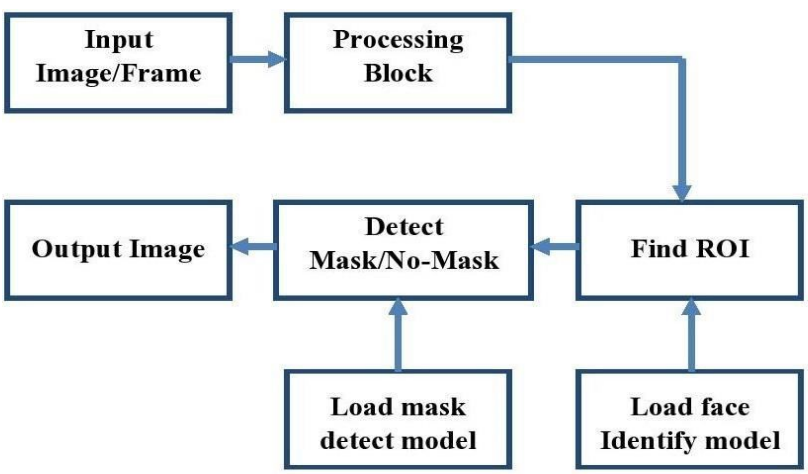

Fig-3:Block Diagram

In order to build a face identification system, the existing face identification model is used. This proposed Project load the model using deep neural network module. The input is taken from the camera and the video is captured, the video is divided into frames in the model, processing can be done by the final model created by training the datasets using FRCNN and Inception V3 methodology, here we load the model trained from Facemask identification project.
Once we are done with loading models, The process holds same for video and image as every frame in a video is image, hence we understand the process of what happens after image is loaded. Once we have one or more detections, for every detection(face) we run the facemask detection model to find if the face is covered with mask or not.

4) *Operating System* - *Windows 7 SP1 or later (64 Bit):* An operating system (OS) is the system software which manages computer hardware, software resources, and provides services for computer programs. Time-sharing operating systems schedule tasks for efficient use of the system and may also include accounting software for cost allocation of processor time, mass storage, printing, and other resources.

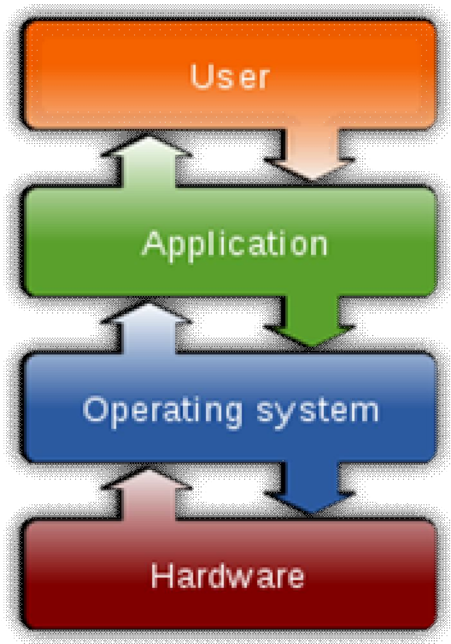

Fig-4: Operating system placement on computer storage

Microsoft Windows is a family of proprietary operating systems designed by Microsoft Corporation and primarily targeted to Intel architecture-based computers, with an estimated 88.9 percent total usage share on Web connected computers. Microsoft Windows was first released in 1985, as an operating environment running on top of MS-DOS, which was the standard operating system shipped on most Intel architecture. Server editions of Windows are widely used. In recent years, Microsoft has expended significant capital in an effort to promote the use of Windows as a server operating system.

5) *Ram – Min 4 GB*: Random-access memory is a form of computer memory that can be read and changed in any order, typically used to store working data and machine code. A random- access memory device allows data items to be read or written in almost the same amount of time irrespective of the physical location of data inside the memory. In contrast, with other direct-access data storage media such as hard disks, CD-RWs, DVD-RWs and the older magnetic tapes and drum memory, the time required to read and write data items varies significantly depending on their physical locations on the recording medium, due to mechanical limitations such as media rotation speeds and arm movement.
6) *ROM – Min 8 GB Free space* :Read-only memory (ROM) is a type of non-volatile memory used in computers and other electronic devices. Data stored in ROM cannot be electronically modified after the manufacture of the memory device. Read-only memory is useful for storing software that is rarely changed during the life of the system, also known as firmware. Software applications (like video games) for programmable devices can be distributed as plug-in cartridges containing ROM.
7) *LabVIEW 2018 or Later (64 Bit)*:Laboratory Virtual Instrument Engineering Workbench (LabVIEW) is a system-design platform and development environment for a visual programming language from National Instruments Module. LabVIEW is commonly used for data acquisition, instrument control, and industrial automation on a variety of operating systems (OSs), including Microsoft Windows as well as various versions of Unix, Linux, and macOS. LabVIEW integrates the creation of user interfaces (termed front panels) into the development cycle. LabVIEW programs-subroutines are termed virtual instruments (VIs). Each VI has three components: a block diagram, a front panel, and a connector pane. The last is used to represent the VI in the block diagrams of other, calling VIs. The front panel is built using controls and indicators. Controls are inputs: they allow a user to supply information to the VI. Indicators are outputs: they indicate, or display, the results based on the inputs given to the VI. The back panel, which is a block diagram, contains the graphical source code. All of the objects placed on the front panel will appear on the back panel as terminals. The back panel also contains structures and functions which perform operations on

controls and supply data to indicators. The structures and functions are found on the Functions palette and can be placed on the back panel. Collectively controls, indicators, structures, and functions are referred to as nodes. Nodes are connected to one another using wires, e.g., two controls and an indicator can be wired to the addition function so that the indicator displays the sum of the two controls. Thus, a virtual instrument can be run as either a program, with the front panel serving as a user interface, or, when dropped as a node onto the block diagram, the front panel defines the inputs and outputs for the node through the connector pane. This implies each VI can be easily tested before being embedded as a subroutine into a larger program.

8) *NI Vision Acquisition Software:* Vision Acquisition Software is driver software for acquiring, displaying, and saving images from a wide variety of camera types. NI Vision Acquisition Software (VAS) enables you to acquire, display, and save images from a range of industry standard cameras interfaces including GigE Vision, USB3 Vision, and Camera Link. You can also use this software to conveniently control digital I/O on NI vision hardware. With a set of easy-to-use functions and example programs, you can quickly create applications using LabVIEW, LabVIEW NXG, and C/C++.
9) *Python 3.6 or Later* :Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.[30] Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.
10) *TensorFlow 1.4 or Later* :TensorFlow is a free and open-source software library for machine learning. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. TensorFlow is a symbolic math library based on dataflow and differentiable programming. It is used for both research and production at Google. TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache License 2.0 in 2015.
11) *Labelling*: Labelling is a free, open-source tool for graphically labelling images. It's written in Python and uses QT for its graphical interface. It's an easy, free way to label a few hundred images.
12) *CUDA:*CUDA stands for Compute Unified Device Architecture it is a parallel computing platform and application programming interface (API) model created by Nvidia.[1] It allows software developers and software engineers to use a CUDA-enabled graphics processing unit (GPU) for general purpose processing – an approach termed GPGPU (general-purpose computing on graphics processing units). The CUDA platform is a software layer that gives direct access to the GPU's virtual instruction set and parallel computational elements, for the execution of compute kernels.
13) *Cudnn:* The NVIDIA CUDA Deep Neural Network library (cuDNN) is a GPU-accelerated library of primitives for deep neural networks. cuDNN provides highly tuned implementations for standard routines such as forward and backward convolution, pooling, normalization, and activation layers. Deep learning researchers and framework developers worldwide rely on cuDNN for high- performance GPU acceleration. It allows them to focus on training neural networks and developing software applications rather than spending time on low-level GPU performance tuning. cuDNN accelerates widely used deep learning frameworks, including Caffe2, Chainer, Keras, MATLAB, MxNet, PaddlePaddle, PyTorch, and TensorFlow.
14) *Microsoft Visual Studio* :Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.
15) *Computer Monitor:* A computer monitor is an output device that displays information in pictorial form. A monitor usually comprises the visual display, circuitry, casing, and power supply. The display device in modern monitors is typically a thin film transistor liquid crystal display (TFT-LCD) with LED backlighting having replaced cold-cathode fluorescent lamp (CCFL) backlighting. Previous monitors used a cathode ray tube (CRT). Monitors are connected to the computer via VGA, Digital Visual Interface (DVI), HDMI, Display Port, USB-C, low-voltage differential signalling (LVDS) or other proprietary connectors and signals.
16) *Web Cam (for real time)* :Webcam is a video camera that feeds or streams an image or video in real time to or through a computer to a computer network, such as the Internet. Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video. Webcam software enables users to record a video or stream the video on the Internet. As video streaming over the Internet requires much bandwidth, such streams usually use compressed formats.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

IV.FLOWCHART

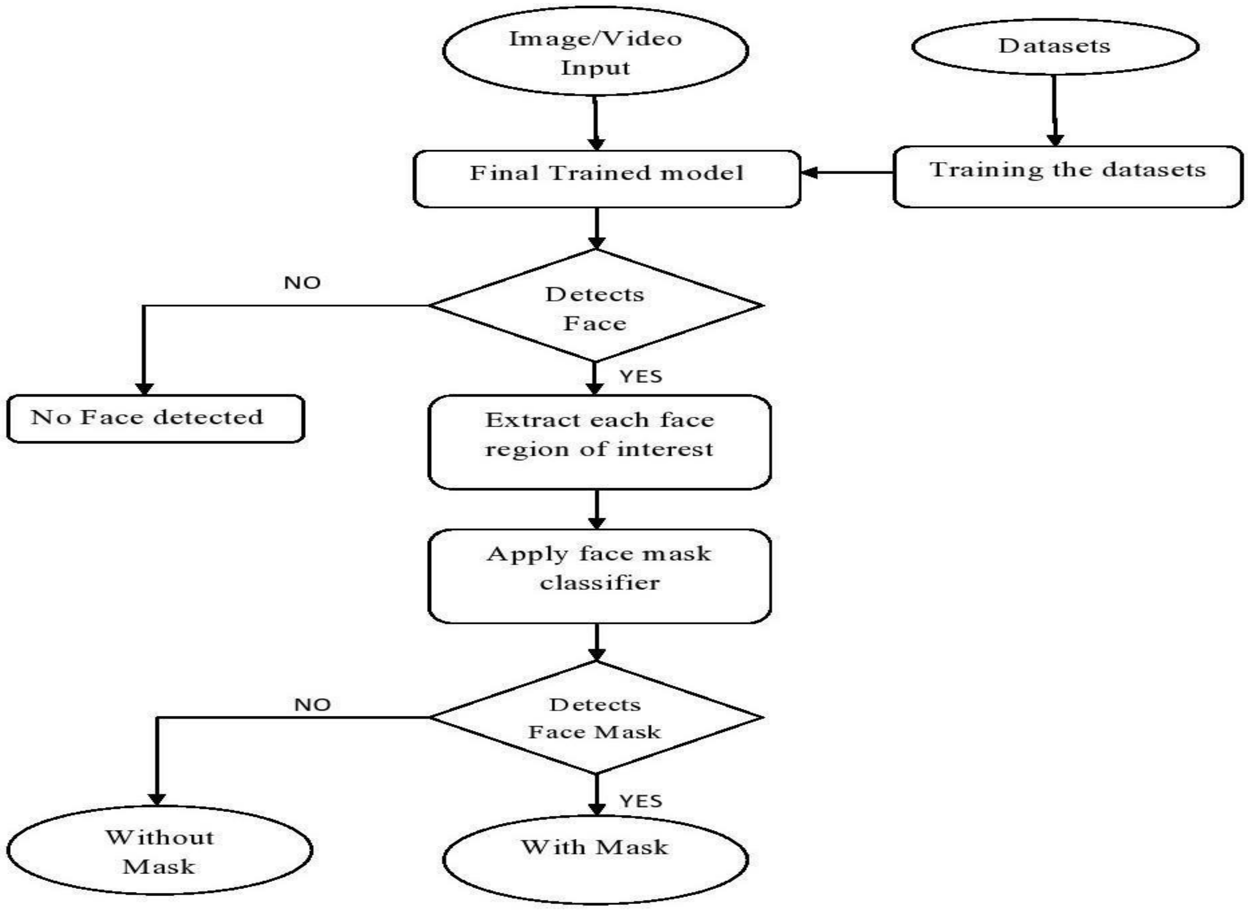

Fig-5: Flowchart of facemask identification using AI-Deep learning neural network

Training the model and freezing the model takes several steps which are first we need to collect the datasets. A data set is a collection of data. In other words, a data set corresponds to the contents of a single database table, or a single statistical data matrix, where every column of the table represents a particular variable, and each row corresponds to a given member of the data set in question. In our project datasets are the images with many faces wearing masks and without masks, then these datasets are used to train the model and 20% of datasets are used to test the model and 80% of datasets are used to train the model, after the training process TF record is created for the model then it is freezed and deployed as final face mask identification model. The TF Record format is a simple format for storing a sequence of binary records. Protocol buffers are a cross-platform, cross- language library for efficient serialization of structured data. Protocol messages are defined by. proto files, these are often the easiest way to understand a message type.

Deployment means to push changes or updates from one deployment environment to another. When setting up a website you will always have your live website, which is called the live environment or production environment. After deploying the model, the model is finalized and freezed. Freezing a layer in the context of neural networks is about controlling the way the weights are updated. When a layer is frozen, it means that the weights cannot be modified further. Deployed model is used for further, Input image/Video frames are taken from the camera and divided as image frames these images are sent to the trained model, the processing of face detection starts. If it detects the face then it extracts the region of interest and applies the face mask classifier to detect the face mask, if it detects the mask, the frame is shown on the monitor, image occurs with green rectangular box tagged as “with mask”, if face mask is not detected then the image frame on the monitor is shown with the red rectangular box which is tagged as “without mask”.

V. RESULTS

Fig-6: With Mask

Fig-7: Without Mask

The above results are experimental which are conducted in different devices, figure 6 shows the with mask images where it identifies the face with mask, the box will cover the face and shows the tag “with mask” on the left corner of the box, here the final deployment is done using python and also LabVIEW, in figure 6 left image shows the result by python and right image shows the result by LabVIEW.

Figure 7 shows the without mask images where it identifies the face without mask, the box will cover the face and shows the tag “without mask” on the left corner of the box, and in above figure 6.2 left image is result from python deployment and right image is result from LabVIEW deployment.

VI. CONCLUSION

As the COVID-19 situation is on a surge around the world, people need to take precautions. Our model is for the safety of public. The face mask identification technology is useful in many public places like Bus stop, railway stations, colleges and schools etc. By using some thesis and based on experimental results we will be able to detect face mask more precisely and identify the face mask individually with exact location in the picture in x, y axis. Papers we referred also identification and compare each method for their efficiencies.

REFERENCES

[1] Agarwal, S., Awan, A., and Roth, D. "Learning to detect objects in images via a sparse, part- based representation”. IEEE Trans. pp. 1475–1490. Volume 1. Year 2004.
[2] B. Alexe, T. Deselaers and V. Ferrari, "What is an object?" IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, pp. 73-80, Year2010.
[3] Aloimonos, J., Weiss, I., and Bandyopadhyay, A. Active vision. Int. J. Comput. Vis. 1, pp. 333– 356, Year 1988.
[4] Andreopoulos, A., and Tsotsos, J. K. “50 years of object recognition: directions forward.Comput. Vis. Image Underst. 117, pp. 827–891. Year 2013.
[5] Azzopardi, G., and Petkov, N, “Trainable cosfire filters for key point detection and pattern recognition”. IEEE Trans. pp. 490– 503. Year 2013.
[6] Azzopardi, G., and Petkov, N. “Ventral-stream-like shape representation: from pixel intensity values to trainable object-selective cosfire models”. Year 2014.
[7] K. L. Masita, A. N. Hasan and S. Paul, "Pedestrian Detection Using R-CNN Object Detector," 2018 IEEE Latin American Conference on Computational Intelligence (LA- CCI), Gudalajara, Mexico, pp. 1-6. Year 2018.
[8] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017.
[9] Krizhevsky A, Sutskever I, Hinton G (2012) ImageNet classification with deep convolutional neural networks. In: Neural Information Processing Systems (NIPS), pp 1097–1105. Year 2012.
[10] Deng J, Guo J, Zhou Y, Yu J, Kotsia I, Zafeiriou S (2019) Retinaface: single-stage dense face localisation in the wild. Year 2019.