



# IJRASET

International Journal For Research in  
Applied Science and Engineering Technology



---

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

---

**Volume:** 11    **Issue:** III    **Month of publication:** March 2023

**DOI:** <https://doi.org/10.22214/ijraset.2023.49393>

[www.ijraset.com](http://www.ijraset.com)

Call:  08813907089

E-mail ID: [ijraset@gmail.com](mailto:ijraset@gmail.com)

# Digital Forensics Face Detection and Recognition

Vinu Varshith Alagappan<sup>1</sup>, Harshul Vaishnav<sup>2</sup>

University of Padova, Padova, Italy

**Abstract:** Face recognition is a biometric system used to identify or verify a person who appear in a scene often captured by security cameras like at Airports, Offices, Universities, Banks, etc. In this project, we have built a Face detection and Recognition model over images and videos. We have used Viola Jones Algorithm to detect a face(s) in an digital image and Convolutional Neural Network to predict it, for identification or verification. We have extended this model to be able to detect and predict faces which are partially covered with glasses, sunglasses, mask and facemask. We have also used data augmentation to increase the dataset size which in result has boosted the performance of the model.

**Keywords:** CNN, Face Recognition, Face Detection, Viola Jones Algorithm

## I. INTRODUCTION

Face detection is the task of localizing a face in an image or video sequence[1]. We have done it using computer vision algorithm, Viola and Jones [2]. Face recognition is the task of classifying a face whenever it has been localized, i.e., telling who is in that face. Face recognition consists of 5 steps, Figure 1. Firstly, the face is localised using Viola-Jones algorithm and is isolated. The face is then preprocessed and augmented. The colorspace is converted to Grayscale<sup>1</sup>. Finally, we match it with the ones against other matches in the database. It can be used for Verification where a face is matched against the previously enrolled template (commonly used in Mobile devices) and for Identification where a face is matched against the entire database (commonly used by Law Enforcement).

We have extended the Lab Experience 4 of the course [1]. We have used the same dataset to train which was used in the Lab 4. It contains a total 29481 images of 13 celebrities, labelled from 0 12. We have extended the AlexNet model [3] which is based on Convolutional Neural Network (CNN). We have implemented the entire system using python3 programming language. Implementation is further discussed later.

<sup>1</sup> <https://stackoverflow.com/a/12761601>

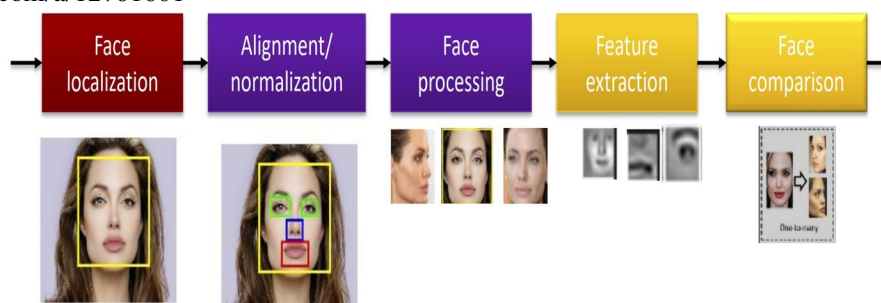


Fig. 1: Face Recognition Steps [4]

The rest of the report is organised as follows. Section 2 describes our model used for Face Detection and Recognition. Implementation and results of the model are presented in Section 3. Section 4 concludes the report.

## II. OUR MODEL

This section briefly describes the algorithms and models we have used for Face detection and recognition.

### A. Viola-Jones Algorithm

In 2001, Viola and Jones [2] proposed an algorithm for rapid object detection. It detects the object using Haar-like cascade classifier.

It consists of the following steps, Figure 2:

- 1) *Haar Features*: Viola and Jones used Haar features like rectangle features.
- 2) *Integral Images*: It is an image representation that stores the sum of the intensity values above and to the left of the image point. This representation allows rectangular feature responses to be calculated in constant time.
- 3) *AdaBoost*: It is used to select the best set of rectangular features.
- 4) *Cascade Classifier*: Instead of applying all 200 filters at every location in the image, train several simpler classifiers to quickly eliminate easy negatives, Figure 3.

### B. Convolutional Neural Network

Convolutional neural networks provides a more scalable approach to image classification and object recognition tasks, leveraging principles from linear algebra, specifically matrix multiplication, to identify patterns within an image. Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs [6].

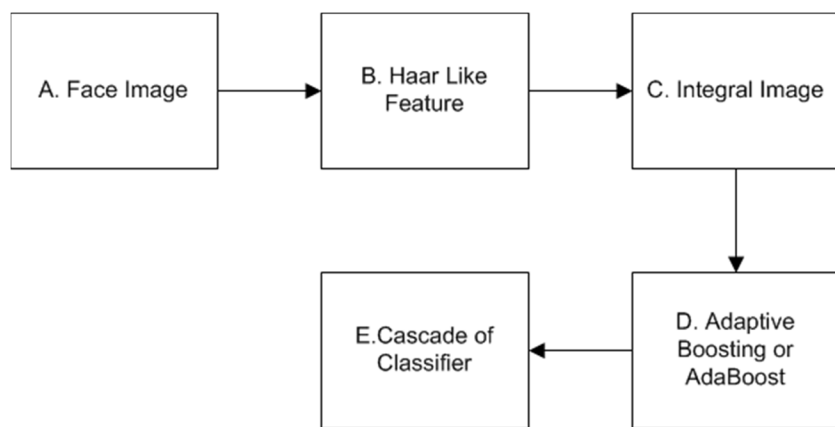


Fig. 2: Viola Jones Steps [5]

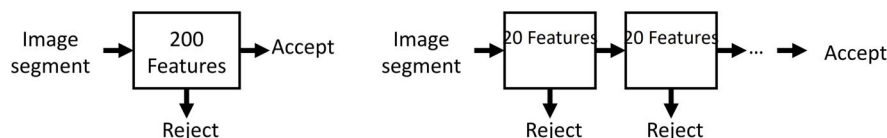


Fig. 3: Cascade classifier [1]

For spatial data like image, this complexity provides no additional benefits since most features are localized. For face detection, the areas of interested are all localized. Convolutional neural networks apply small size filter to explore the images. The number of trainable parameters is significantly smaller and therefore allow CNN to use many filters to extract interesting features.

The CNN model is shown in figure 4. It involves the following layers:

- 1) Convolutional layer
- 2) Pooling layer
- 3) Fully-connected (FC) layer

### C. Dataset

We have used the training dataset of the Lab 4: Face Recognition which contains a total 29481 images of 13 celebrities, labelled from 0 to 12. We have used augmentation to increase the dataset size to 88456. It greatly improves the performance of our model, discussed later. Dataset size can be further increased to improve the performance of the model, but due to limited memory we have used the dataset, Table 1.

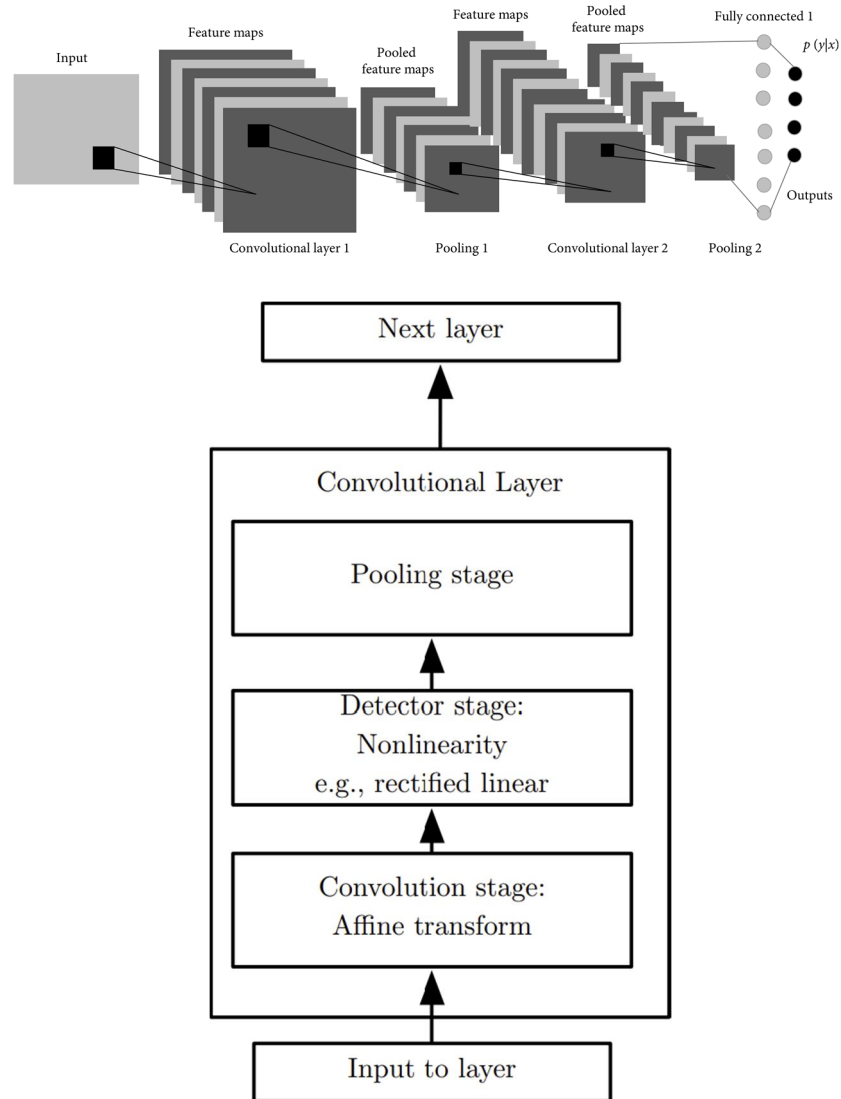


Fig. 4: CNN Model [7], [8]

Label	Celebrity Name	Number of Images	Augmented Images
0	Adam Sandler	1753	5260
1	Alyssa Milano	1906	5719
2	Bruce Willis	1503	4510
3	Denise Richards	2287	6862
4	George Clooney	3078	9235
5	Gwyneth Paltrow	2949	8848
6	Hugh Jackman	2091	6274
7	Jason Statham	992	2977
8	Jennifer Love Hewitt	1896	5689
9	Lyndsay Lohan	7752	23257
10	Mark Ruffalo	1066	3199
11	Robert Downey Jr	711	2134
12	Will Smith	1497	4492

Table 1: Dataset used for training the model

### III. IMPLEMENTATION AND RESULTS

We have used the algorithms and models mentioned in previous section to create an application for Face detection and recognition over images and videos. Furthermore, it is also able to recognise a face with face mask. We have two python files one to train the model and other one to test the model. Figure 5, shows the performance of training the model. Table 2, shows the libraries used.

Firstly we have create and trained our CNN model over the given training dataset. 20% of the training dataset is used for the validation set and a separate set is used for the testing of the model which contains images which are not used for the training. We have extended the AlexNet model by adding additional parameters to improve the performance of our model. Our CNN model can be visualised<sup>2</sup> by Figure 6.

For the face detection, we have used Viola Jones algorithm which is already implemented in the OpenCV library [15]. With OpenCV, we detect the face in the image or video and create a rectangular box around it, shown in Figure 7a. The trained model is then used to predict the face, detected by the Viola Jones algorithm, Figure 7b.

<sup>2</sup>PlotNeuralnet [9] is used to generate the model image.

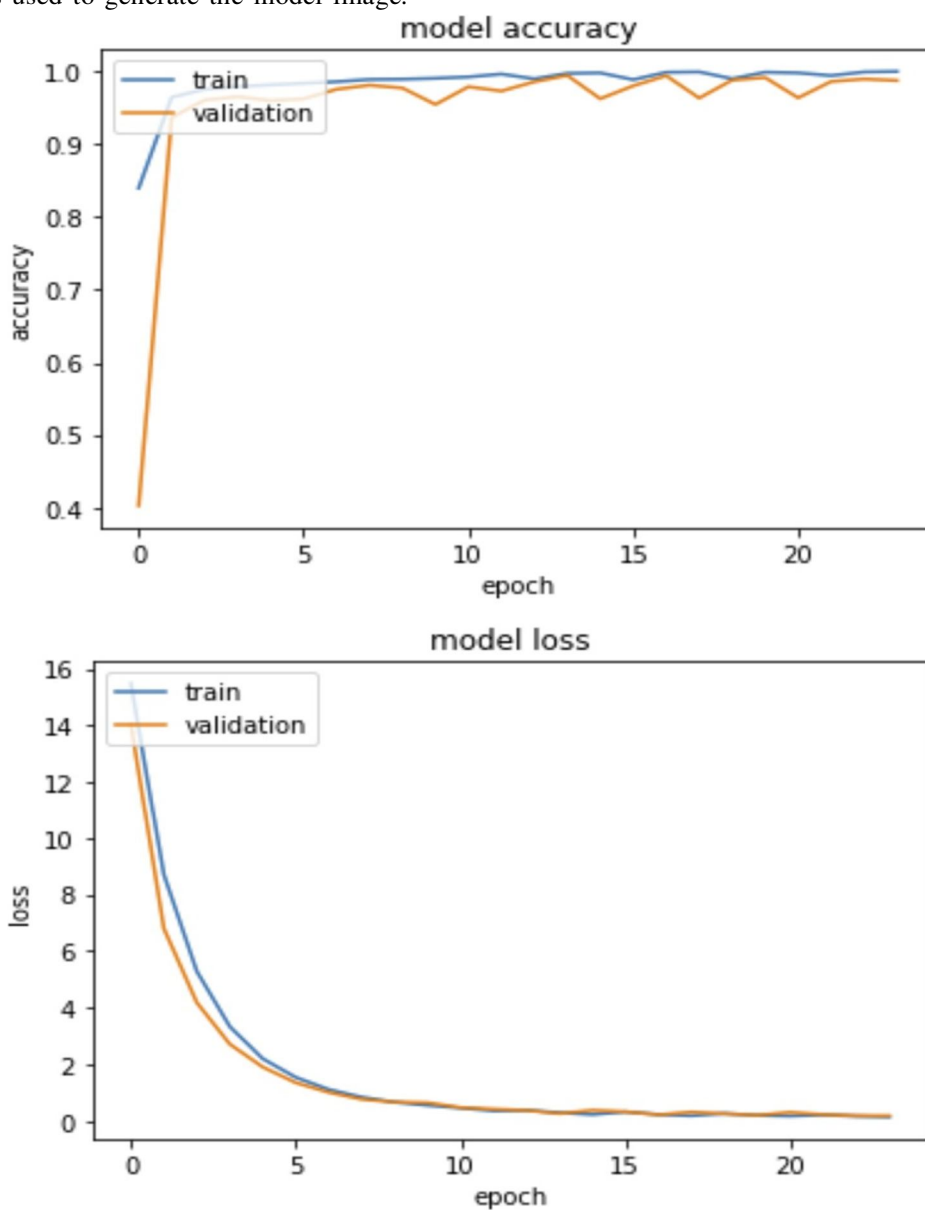


Fig. 5: Model Training



Name	Use
Tensor Flow [9]	A Machine Learning Library
sys, os, random	Python support libraries
Numpy [10]	Library used to deal with large arrays, algebra, etc
Matplotlib [11]	Library for Visualisation
sklearn [12]	Used for confusion matrix and shuffling the dataset
Pandas	Data Manipulation Library (Used for confusion matrix)
seaborn [13]	Data Visualization library based on matplotlib
Pillow [14]	Python Imaging Library
OpenCV [15]	Computer Visioin Library (used for detecting and processing face)
Face-recognition [16]	Face Recognition library (Used only for face detection)

Table 2: Libraries used for the Face Recognition

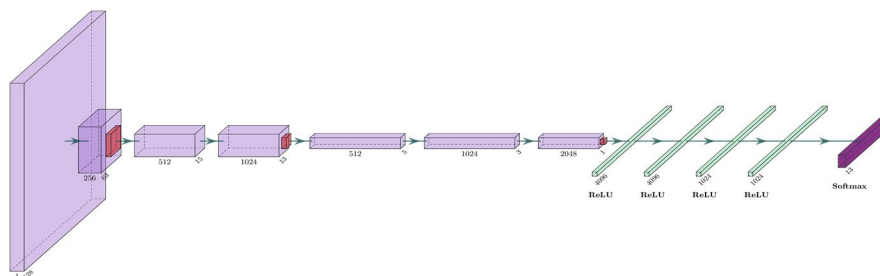


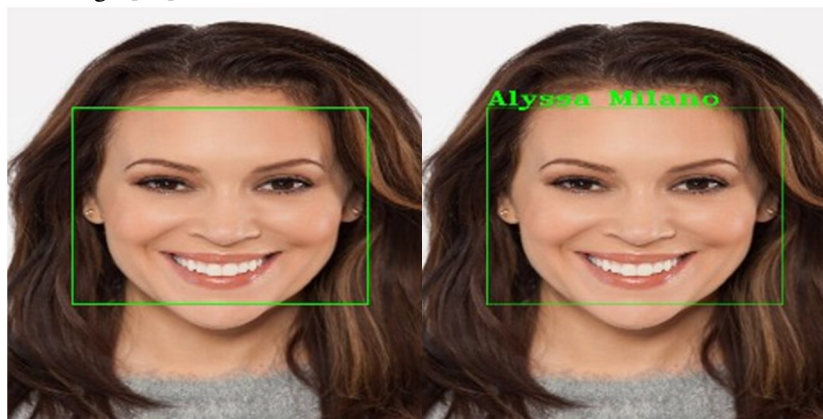
Fig. 6: Our CNN Model

To mask the face, we have used the MATLAB code provided to us with the dataset to apply mask over the images. The output of the MATLAB code is shown in Figure 7c. With the above CNN model, we are able to correctly predict the masked face, Figure 7d. Similarly, for the Face Mask, we have used an external program<sup>3</sup>, Figure 7e which is also correctly predicted by our model, Figure 7f.

We have extended our dataset by using Data Augmentation (with external library<sup>4</sup>) to increase the size of dataset which in result increases the performance of the model. Table 1 shows the size of augmented dataset. Table 3, shows the performance of our model over different datasets. It can be seen that the augmented dataset performs better than the normal dataset.

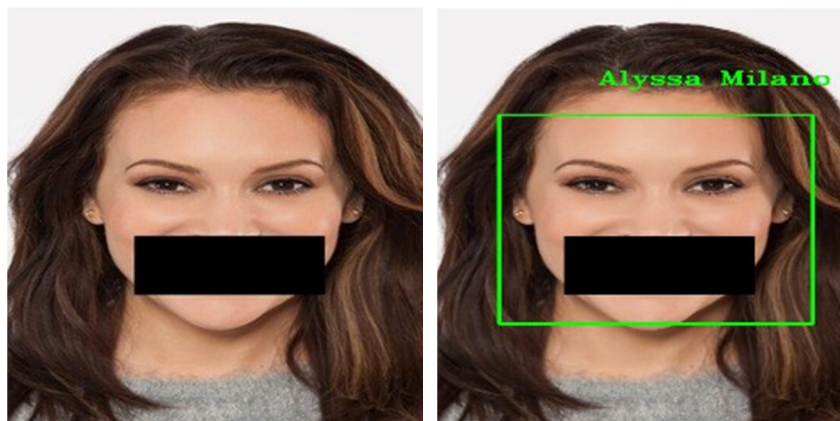
<sup>3</sup>Python program to add a face mask over the face [18]

<sup>4</sup>To create obstacles over the image [19]



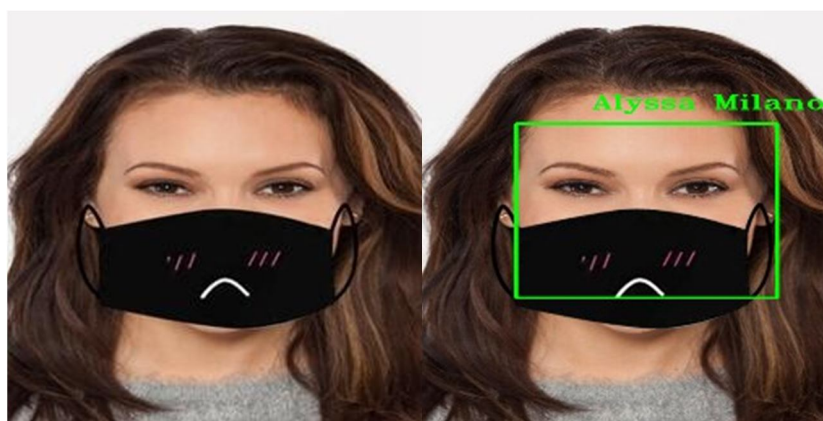
(a) Detected Face

(b) Recognised Face



(c) Masked Face

(d) Recognised Masked Face



(e) Facemasked Face

(f) Recognised Facemasked Face

Fig. 7: Input and Output of the CNN Model

Test Set	Accuracy	
	Normal Training Set	Augmented Training Set
Normal	97.57%	98.09%
Glasses	76.65%	88.34%
Sun Glasses	49.18%	69.66%
Mask	21.99%	96.82%
Facemask	39.39%	75.87%

Table 3: Performance of the model

Face recognition on videos is same as recognising face on images. Videos are formed by a number of still images, called frames. So, initially we take a frame from a video and predict the face detected on that frame and display the result. Looping through all the frames, if we can predict the result faster than the fps, then we can show the images with the original fps of the video in real time. Otherwise, the real time video will be slowed, meaning lower fps.

#### IV. CONCLUSION

The accuracy over the different test sets can be further improved using augmentation. We aren't able to increase the augmentation dataset size due to limited memory. In some cases, OpenCV wasn't able to detect a face in an image, we have solved this issue with the face-recognition library. We also observed that complexity of the CNN model affects the performance on predicting the face or recognising the face. Our model and python code can also be used to detect and predict multiple faces in an image or video.

## REFERENCES

- [1] Course - Digital Forensics 2021/2022, <https://elearning.dei.unipd.it/course/view.php?id=8610>.
- [2] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." In Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, vol. 1, pp. I-I. Ieee, 2001.
- [3] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information process-ing systems 25 (2012).
- [4] Course - Biometrics 2021/2022, <https://elearning.unipd.it/math/course/view.php?id=933>.
- [5] Effendi, T. M., H. B. Seta, and T. Wati. "The combination of viola-jones and eigenfaces algorithm for account identification for diploma." In Journal of Physics: Con- ference Series, vol. 1196, no. 1, p. 012070. IOP Publishing, 2019.
- [6] Convolutional Neural Networks, <https://www.ibm.com/cloud/learn/convolutional-neural-networks>
- [7] Zhao, Hongbo, and Bingrui Chen. "Data-driven model for rockburst prediction." Mathematical Problems in Engineering 2020 (2020).
- [8] Course - Deep Learning 2021/2022, <https://elearning.unipd.it/math/course/view.php?id=885>
- [9] Abadi, Mart´in, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado et al. "Tensorflow: Large-scale machine learning on hetero- geneous distributed systems." arXiv preprint arXiv:1603.04467 (2016).
- [10] Harris, Charles R., K. Jarrod Millman, St´efan J. Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser et al. "Array programming with NumPy." Nature 585, no. 7825 (2020): 357-362.
- [11] Hunter, John D. "Matplotlib: A 2D graphics environment." Computing in science engineering 9, no. 03 (2007): 90-95.
- [12] Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel et al. "Scikit-learn: Machine learning in Python." the Journal of machine Learning research 12 (2011): 2825- 2830.
- [13] Waskom, Michael L. "Seaborn: statistical data visualization." Journal of OpenSource Software 6, no. 60 (2021): 3021.
- [14] Clark, Alex. "Pillow (pil fork) documentation." readthedocs (2015).
- [15] Bradski, Gary. "The openCV library." Dr. Dobb’s Journal: Software Tools for the Professional Programmer 25, no. 11 (2000): 120-123.
- [16] Face-recognition Library, [https://github.com/ageitgey/face\\_recognition](https://github.com/ageitgey/face_recognition)
- [17] PlotNeuralNet, <https://github.com/HarisIqbal88/PlotNeuralNet>.
- [18] Face-Mask, <https://github.com/Prodesire/face-mask>.
- [19] Partial-Face images classification model using CNN, <https://github.com/PongC/Partial-Face-Recognition>





10.22214/IJRASET



45.98



IMPACT FACTOR:  
7.129



IMPACT FACTOR:  
7.429



# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24\*7 Support on Whatsapp)