FACE RECOGNITION

Submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

in

Computer Science and Engineering

by

Dhruv Mittal

17BCE2110

Under the guidance of

Prof. Manjula R

SCOPE

VIT, Vellore.

June, 2021

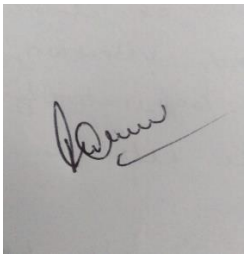
DECLARATION

I hereby declare that the thesis entitled "Face Recognition" submitted by me, for the award of the degree of Bachelor of Technology in Computer Science and Engineering to VIT is a record of bonafide work carried out by me under the supervision of Prof. Manjula R.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place : Vellore

Date : 6th June 2021



Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled "Face Recognition" submitted by Dhruv Mittal & 17BCE2110, School of Computer Science and Engineering, VIT, for the award of the degree of Bachelor of Technology in Computer Science and Engineering, is a record of bonafide work carried out by him / her under my supervision during the period, 01.02.21 to 06.06.2021, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfills the requirements and regulations of the University and in my opinion meets the necessary standards for submission.

Place : Vellore

Date : 6th June 2021

Signature of guide

Internal Examiner **External Examiner**

Head of the Department
Computer Science and Engineering

ACKNOWLEDGEMENTS

In this capstone project, I went to great lengths and worked really hard. However, it would not have been possible without the supervision of my mentor Prof. Manjula R in this Capstone Project. I'd like to express my profound gratitude from the bottom of my heart. Also, I would like to express my gratitude towards her as she has been very patient and committed assistance during my capstone project.

I am highly indebted to School of Computer Science and Engineering for giving me an opportunity and their supervision and providing necessary information for this research and development.

I am very thankful to my parents, my all time supporters for their co-operation and encouragement during the final year of my graduation as COVID has impacted lives of so many people directly or indirectly.

Dhruv Mittal

Executive Summary

In modern security system sector, face identification techniques have been the most significant biometrics system. Face recognition system verifies the humans by unique characteristic traits and fastest bio-metric technology. Our society has shown an increase in demand of security biometrics and advancement in technology, extracting and accessing the identity information has become simpler.

I have been motivated to build this project related to face recognition because it has become easier to track down criminals, thieves and trespassers etc. This technology can be integrated with surveillance cameras and CCTV cameras in important places such as institutes, corporates, government buildings etc.

This application is not only limited to security needs and tracking down suspected people, but also can used for any unidentified persons, it could also make it easier to seek out missing children and seniors.

Our project aims on building a face recognition system which will be able to identify the persons in front of the camera with their identity. Comparing traditional methods of ML algorithms and DL techniques, the latter outperformed very well if terms of high prediction and robustness.

Our methodology is first to collect the images as dataset, which we will be doing Face detector program using OpenCV which captures the faces detected and store it desired location and then classify the dataset into training and testing dataset. Once we complete our image collection for dataset, we can use this dataset to for Face recognition program which will generate a model and tell us the accuracy of the proposed model. After that, we use an interface which will capture image in front of camera and reveal the identity of the person. This is how are face recognition system will work.

CONTENTS

List of Figures

List of Tables

List of Abbreviations

CNN	Convolutional Neural Network
DP	Deep Learning
AI	Artificial Intelligence
OS	Operating System
RAM	Random Access Memory
UI	User Interface

1. INTRODUCTION

1.1 THEORITICAL BACKGROUND

The examination in biometric strategies and procedures for perceiving people upheld their social and actual attributes or characteristics, face acknowledgment is an acquired quality and it's something that people for the most part perform easily and regularly in their way of life and it's the technique for recognizing people from their appearances natural attributes. Researchers in the fields of biometric based systems, design recognition, field of computer vision, and ML are focusing on automated face recognition as one of their key focuses and objectives of research.

Face identification may be a difficult pattern identification problem in computing, despite the fact that humans can recognise faces without much effort. Facial recognition will attempt to recognise a person's face, which is a three-dimensional object that changes appearance depending on lighting and countenance, and is supported by a two-dimensional photograph.

Face recognition algorithms take landmarks, or features, from an image of the subject's face to identify countenance. Face recognition systems, for example, check for human characteristics such as eyes, nose, jaw shape, lips, and so on. These characteristics are then used to find more photos with similar characteristics. One of the earliest successful systems is predicated on template matching techniques applied to a group of salient countenance, providing a kind of compressed face representation.

1.2 MOTIVATION

Criminals, thieves, and trespassers are easier to catch with face recognition. For surveillance and monitoring, the technology might be integrated into CCTV cameras.
Forensics teams can use this technology to conduct criminal investigations or to determine someone's identity just based on his face photograph.
The technology isn't just for catching crooks. It may, for example, make it easier to locate missing children and elders. This type of prototype might be utilised in workplaces and even universities for attendance systems that detect faces and mark an individual's presence automatically. Employees do not have to validate their identities or clock in with plastic cards because this method is speedier than traditional roll call systems.

When known shoplifters, retail thieves, or those with a history of fraud enter retail outlets, face recognition technologies can be utilised to rapidly identify them. When a customer enters a shopping mall or supermarket, their faces might be matched against massive databases of offenders, allowing the loss of prevention and supermarket security personnel to be quickly warned.
Face recognition is currently used to unlock a range of phones, including the latest Android and Apple handsets. This technology has shown to be a great tool to protect personal data and guarantee that sensitive data stays unavailable to the criminal if a phone is stolen or misplaced.
These all are the applications of Face recognition projects which motivates us to build a project related to Facial detection and recognition.

1.3. Aim of the proposed Work

This project seeks to create a face detection and recognition program. The most basic goal is to recognize the person and fetch information from a database.

Our aim in this project is to first detect face and compare it with the faces in our database and then recognize the person.

We intend to build a system which has many applications in future as stated above with high accuracy.

We aim to build such a system that will be able to detect a person's face and store it as a dataset, and in future can be used to identify this person using face recognition system.

1.4. Objective of the proposed work

Our primary objective is to detect the faces in the live camera using OpenCV library and store it as a dataset. Here we are detecting the faces and creating the dataset for our project at the same time.

After the dataset collection is done, we will use Transfer Learning techniques to train the images and our model and use this model further to create a User Interface which would detect the face in front of camera actually tell us the name of the of the person in front of web cam.

We will be using the Transfer Learning technique of InceptionResNetV2 model provided by Keras Library.

2. Literature Survey

2.1. Survey of the Existing Models/Work

Table 2.1 Literature survey

PAPER	OBJECTIVE OF PAPER	ALGORITHM AND METHODOLOGY	FUTURE SCOPE
"Human face detection algorithm via Haar cascade classifier combined with three additional classifiers" (2017)	To use the Haar cascade classifier with skin hue histogram matching, eyes detection, and mouth detection to deploy a human face recognition algorithm.	A new facial detection technique is planned that combines three weak classifiers that are entioned in the objecive to create a more robust complete detection system. **Result:** Faces detected by Haar classifier: 78.18% Faces detected by the proposed system: 98.01%	More research should be done in the future on detection for individuals of several skin colors, rather than of single or similar color faces, as we did in the planned model.
"Comparison between attendance system implemented through haar cascade classifier and face recognition library" (2021)	The proposed paper and model's major goal is to improve the traditional method of registering attendance and replace it with face recognition biometrics.	**Haar Cascade Model:** The model requests the child's registration id and contact information. It saves all of this information in a CSV or MS Excel file, and then uses an integrated camera to capture a facial image. This face image is maintained in the dataset and is connected to the registeration id and contact number. Following the creation of the database, it is utilised to automatically mark the attendance of the students. **Face Recognition Library**: Following the capture of the student's image, the registration id and contact information are entered as inputs. These photos are now kept in a dataset with registration id and contact information. If the kid is present in school after gathering all of the data, his or her attendance will be automatically marked	By including the iris detection approach into the model, the model's accuracy will be improved.

		using the this face recognition system. The photos from the camera footage will be compared to those in the dataset. If the algorithm recognizes the face, it will mark the attendance of the student against the the school roll number. **Result:** The suggested approach benefits from Haar-like qualities since the model is less time exhaustive, more accurate, and effective.	
"Face Recognition and Identification using Deep Learning Approach" (2006)	The goal is to show how to use OpenCV in Python to build and create a face recognition application utilising DP.	Faces are recognised using Haar feature-based cascade classifiers after images are provided as input and recorded in the dataset using a camera sensor. After the classifier has been trained, the TensorFlow framework and CNN classifier are used to begin identifying the faces. **Result :** The system's over-all accuracy for face recognition from the generated model is **91.7%.**	More training images recorded in low light intensity can be included in the model to improve prediction.
"Face Recognition Based on Convolutional Neural Network by Musab Coşkun and Ayşegül Uçar" (2017)	The purpose of this research was to create an algorithm that would be effective at recognising faces, that is also accurate. Adding a Batch Normalization process after two separate layers for the training step to produce a new CNN architecture.	It starts with the pre-processing phase, which involves colour space conversion and image scaling, then moves on to facial feature extraction and categorization. **Result:** Overall Prediction accuracy obtained is 92%.	The authors of this paper haven't mentioned about the future scope or any improvements in this research.
"Illumination-robust face recognition based on deep convolutional neural networks architectures"(2020)	To compare the face recognition performance of Deep CNN models such as VGG16, ResNet50, and Inception-V3	The face recognition system is broken down into three phases: (1) face picture enhancement, (2) Application of Voila	The editors of this paper haven't mentioned about the future scope or any improvements in this research.

	Convolutional Neural Networks (CNN).	Jons method for face identification, and (3) extraction and classification of facial features. **Result:** VGG=97.28% ResNet50=98.35% InceptionV3=99.4%	

2.2. Summary/ Gaps identified in the Survey

Table 1.2 Summary and limitations of Papers

RESEARCH PAPER	SUMMARY / GAP
"Human face detection algorithm via Haar cascade classifier combined with three additional classifiers" (2017)	Because the histogram was trained from specific photographs of humans, the model of the trained set's histograms should generalise to other faces of individual's as well, till we continue to test the human faces that are of same race as training images.
"Comparison between attendance system implemented through haar cascade classifier and face recognition library" (2021)	The main significant drawbacks of utilising the Haar cascade classifier is that it is ineffective at detecting tilted faces. The model will be not be much accurate if angle of face not tilted by 90 degree, like 90, 180, 270 or 360.
"Face Recognition and Identification using Deep Learning Approach" (2006)	The well-lit settings might be considered as a characteristic that effects the face recognition procedure based on the results. When the light concentration is low, the recognition system has a leaning to make errors with facial recognition.
"Face Recognition Based on Convolutional Neural Network by Musab Coşkun and Ayşegül Uçar" (2017)	This research will result in a face recognition model assessment utilising the CNN algorithm. The suggested system's key notion is that it uses batch normalisation for the result of the first and last convolutional layers, resulting in improved accuracy.
"Illumination-robust face recognition based on deep convolutional neural networks architectures" (2020)	They proposed a face detection based on the Viola Jones method, improved facial distinction, and 3 deep CNN architecture. The VGG16, ResNet50, and InceptionV3 CNN architectures were used in their experiment. The proposed approach outperformed all others when compared to other approaches.

3. Overview of the Proposed System

3.1. Introduction and Related Concepts

OpenCV Library- A set of software development features aimed at making real-time computer vision applications possible. The OpenCV library functions were created to provide a standard foundation for the software based on computer vision technology and to speed up the perception of machine applications in commercial business.

It's a library that helps us know how photos and videos are saved in the dataset, as well as how to inspect and extract data. Computer Vision is the foundation or most commonly utilized technology in AI. It's used in self-driving cars, robotics, and photo editing apps, among other things. Its being used in automated cars with auto pilot features, picture editing, application in robot manufacturing, picture editing and many other.

This library has many optimized algorithms and are often used in projects of face detection, in medical technology, manipulation with images and retrieving data from video and images. Also used in tracking systems such as camera movements tracking of objects. Many other implementations like combining different parts of same images. We would have often noticed red eyes in photos that are caused due to flashlight of the camera, OpenCV algortihms are used in the photo editing software and mobile apps.

Transfer Learning- Transfer learning is a procedure wherein a model made for one errand is used as the reason for a model on an alternate assignment. Because the "transfer" of information from one "parent" model to a "child" model allows the "child" model to be trained to high accuracies with a much smaller dataset than the "parent" model, transfer learning is very popular in Deep Learning.

Keras Applications are DP models that come with weights that have already been trained. Prediction, feature mining, and fine-tuning are all common applications for these models.

Transfer Learning takes the knowledge gained from one problem and applies it to other problems that are connected to the first.

The main concept is to apply what a model has learnt after training over millions of objects with thousands of objects to new model with less data. Rather of starting from the beginning, we start with patterns discovered while completing a related activity.

Transfer learning is quite different form the machine learning technique, although it is frequently referred to as a "design methodology" in the industry. It's also not a sub-discipline, sub part or research area of machine learning. Nonetheless, it's now quite general in conjunction with neural networks, which require a lot of data and computer capacity.

3.2. Framework, Architecture or Module for the Proposed System (with explanation)

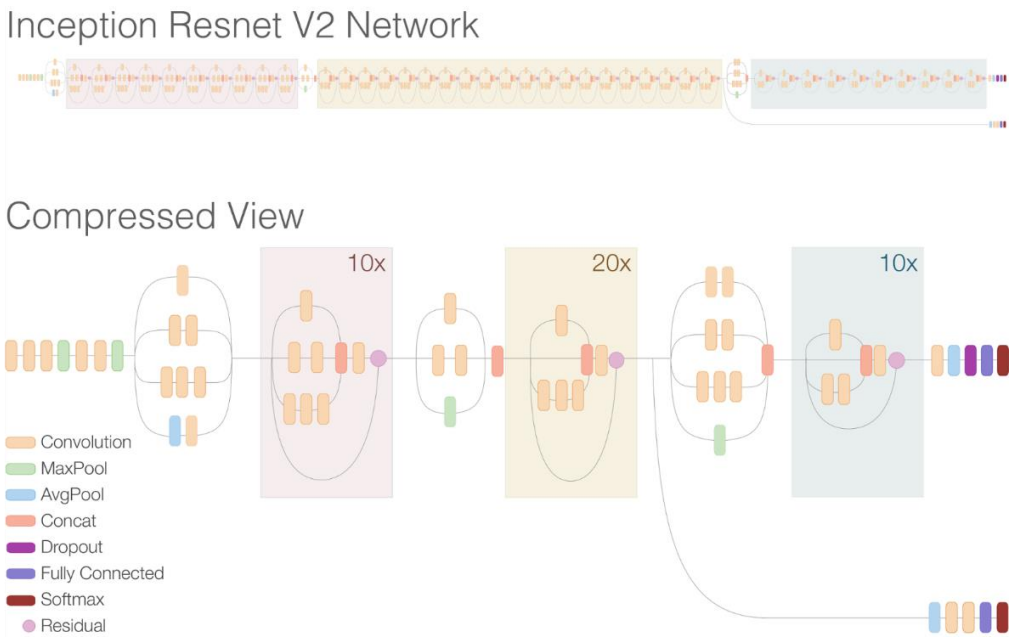


Figure 3.1 InceptionResNetV2 Model

As we can see this is a compressed view of InceptionResNetV2 which comprises of several layers and has 572 layers of depth. This is a model and application provided by keras library. It's a pre trained model used for the famous database called "ImageNet" for image classification, where it could classify thousand different objects.

The Convolutional Neural Network (CNN) Inception-ResNet-V2 was trained over 1,000,000 photos from the ImageNet database.

With remarkable performance at a comparatively low computational cost, ResNet and Inception are critical to the most major breakthroughs in image recognition performance in recent times.

3.3. Proposed System Model (ER Diagram/UML diagram/Mathematical Modeling)

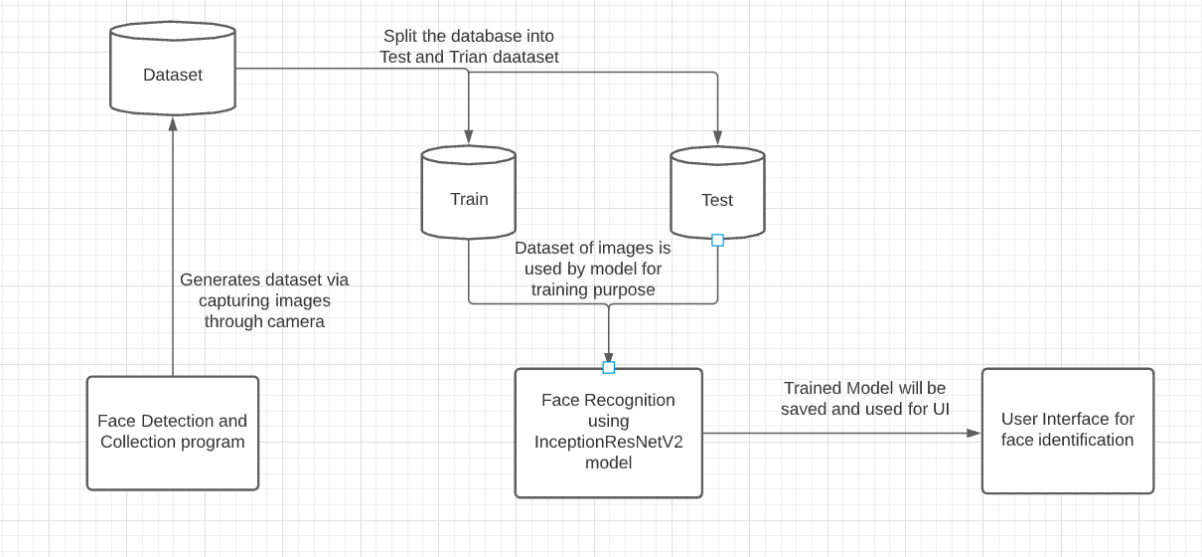


Figure 3.2 Proposed System Model

This is the proposed system model of our project presented above.

Our proposed system depends on the data collection through face detection program and face recognition algorithms used.

Firstly, we will start by collecting data for the dataset through face detection technique and store the captured images in the desired location.

We will split the captured images into training dataset and testing dataset. Moreover, this dataset of training and testing dataset of images is used by Face recognition program using InceptionResNetV2 transfer learning technique which will generate a model which is trained on this dataset.

This trained model will be used for the user interface for facial identification program which will tell identify the person in front of the camera.

4. Proposed System Analysis and Design

4.1. Introduction

Face detection is a crucial field in computer vision that is used in face identification, CCTV integration, criminal investigation, security systems in supermarkets and a variety of other applications.

Face detection and dataset collecting in a specific folder via live web cam using the OpenCV library are the first steps in our proposed system. We now utilise this dataset to manually segregate the Training and Testing photos after the dataset has been prepared.

Now, using the Transfer Learning technique called InceptionResNetV2 offered by the keras package, we can use this well-defined dataset with separate Training and Testing images to train our model. Keras not only allows you to use TensorFlow effectively, but it also assists you in fine-tuning your model.

InceptionResNetV2 is a keras-based image classification model that can classify thousands of different objects for a given model.

The following are the reasons why we chose InceptionResNetV2:

Source: https://keras.io/api/applications/

Model	Size	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth
Xception	88 MB	0.790	0.945	22,910,480	126
VGG16	528 MB	0.713	0.901	138,357,544	23
VGG19	549 MB	0.713	0.900	143,667,240	26
ResNet50	98 MB	0.749	0.921	25,636,712	-
ResNet101	171 MB	0.764	0.928	44,707,176	-
ResNet152	232 MB	0.766	0.931	60,419,944	-
ResNet50V2	98 MB	0.760	0.930	25,613,800	-
ResNet101V2	171 MB	0.772	0.938	44,675,560	-
ResNet152V2	232 MB	0.780	0.942	60,380,648	-
InceptionV3	92 MB	0.779	0.937	23,851,784	159
InceptionResNetV2	215 MB	0.803	0.953	55,873,736	572
MobileNet	16 MB	0.704	0.895	4,253,864	88
MobileNetV2	14 MB	0.713	0.901	3,538,984	88
DenseNet121	33 MB	0.750	0.923	8,062,504	121
DenseNet169	57 MB	0.762	0.932	14,307,880	169
DenseNet201	80 MB	0.773	0.936	20,242,984	201
NASNetMobile	23 MB	0.744	0.919	5,326,716	-
NASNetLarge	343 MB	0.825	0.960	88,949,818	-

Figure 3.3 List of Keras Application Models

This is the table which shows us the different models of the keras library where InceptionResNetV2 model gives us a quite high accuracy with less number of parameters as compared to the NASNetLarge model.

4.2. Requirement Analysis

4.2.1. Functional Requirements

4.2.1.1. Product Perspective

By giving its facial recognition capabilities, this software system of face recognition will ideally give a model that is necessary for biometrics and facial identification in many institutes in our nation. It is python application which is meant for facial biometrics and identification purposes.

4.2.1.2. Product features

This product or application is capable to detect the human faces in front of the webcam in real time, crops the image and stores them in the desired location for dataset collection of images.
Furthermore, it uses transfer learning technique for the model generation after the model is trained with dataset images.
We will be able to identify the persons standing in front of web cam with their names as their identity.

4.2.1.3. User characteristics

The end users and the customers who is using the product should be in a well-lit room condition.
The user should be steady in front of camera and look directly into the camera, not make any kind of movements otherwise it will may

create problems in detecting the faces, resulting less accuracy for face recognition of that particular person.

The user does not have much interaction with the application but his face should be with minimal accessories like there should be no spectacles, sunglasses or any other kind of face mask etc. that would prevent the extraction of facial features.

4.2.1.4. Assumption & Dependencies

This project has several dependencies such as OpenCV library, TensorFlow, Keras, InceptionResNetV2 application from keras library, glob library.

This project fall on the dependencies related to hardware like a powerful GPU required on a local system, RAM is required at least 8GB which will be required if high batch size is considered, an integrated camera of high resolution also required so that the images captures are of much better quality.

4.2.1.5. Domain Requirements

This project is based on Transfer Learning concepts on the application of Face Recognition. Its major domains are facial detection using python programming language and OpenCV library, Facial recognition using the Transfer Learning technique.

Python is the base programming language used in the project along with libraries and its functionalities.

4.2.1.6. User Requirements

The user requirements will be satisfied as they will be able to store them selves in the dataset collection successfully.

The system will deliver the users with face recognition system

which will tell us who is the person standing in front of the camera. This will satisfy the user requirements in a broader field with multiple applications such as security systems, integration with CCTV cameras in buildings, cyber forensics usage for finding missing persons or finding wanted criminals.

4.2.2. Non-Functional Requirements

4.2.2.1. Product Requirements

Product requires a good hardware system with GPU to operate the model training, this project is mainly being operated on Google Colab which provides a much powerful GPU for model training as it would take much less time and much efficient else the local computer system even with a descent dedicated GPU and CPU would take many hours for the model to train.

4.2.2.1.1. Efficiency (in terms of Time and Space)

The face detector program could detect and capture human faces like 200 images at time in about 6-8 sec which is quite fast.
Model training was done on total images of 2100 i.e. Training images=1400, Test images=700, which took around 30-40 min on Google Colab with GPU Nvidia Tesla T4. It consumes around 1.38 GB of RAM and consuming 40GB storage on Google Colab.

4.2.2.1.2. Reliability

Reliability of any software product can always have any kind of bugs or issues and this concept can be difficult to grasp. It could fail in very less conditions, unless the inputs of image into dataset are not appropriate.

We won’t say that that our software product is 100% reliable, of course there could be some faults and loop holes, and also some scope of improvement is always there, but we try our best to keep it reliable for the end users.

4.2.2.1.3. Portability

This system can be deployed and potability of the product can be made by creating the API and exposing it to front end application. Could be deployed to any cloud service like Heroku, Google cloud etc. and portability can be achieved for the software product.

4.2.2.1.4. Usability

This software product is easy to use for the customers as it does not require much human computer interaction (HCI), also the interface is also simple by which the customers should not get intimidated at all. Hence, the goals of the customer are achieved quickly without any hassle.

Face recognition offers a wide variety of applications to contribute to the society. They can be utilised as attendance monitoring systems at universities and business workplaces, as well as being integrated into security cameras as biometrics. They can also be employed by criminal investigation organisations.

4.2.2.2. Organizational Requirements

4.2.2.2.1. Implementation Requirements (in terms of deployment)

Deep Learning models can be deployed using Flask microframework written in python and using cloud services such as Heroku or AWS EC2 Instances.

4.2.2.2.2. Engineering Standard Requirements

Engineering Requirements are that the operator of this Face recognition application should be aware of Python programming language, Machine Learning and Deep Learning Domains.

4.2.2.3. Operational Requirements (Explain the applicability for your work w.r.to the following operational requirement(s))

- **Economic**

  Face recognition software systems can be beneficial to the economy because they can generate demand for the software in the market, which generates cash and contributes to the country's economy.

  According to Markets & Markets, the facial recognition software market is expected to produce $7 billion in sales by 2024.

- **Environmental**

  For this face recognition system to detect the faces properly, the individual should be in a well lightened condition. This is necessary because in dark or less lighting conditions wouldn’t help to detect and recognize the faces.

- **Social**

  This method will raise the bar on monitoring. It enables people to be monitored in an automated and indistinguishable manner. It has the potential to adapt people to new biometrics methods while also increasing the utilisation of technology across society.

- **Political and Legality**

  The police in the state of Uttar Pradesh are utilising an AI-based facial identification system to notify them when a woman is in suffering. The same software was also utilised by the police to deploy surveillance on anti-CAA protests, resulting in over 1,100 arrests.

  This demonstrates that face recognition technologies are lawful, in demand, and may be used for criminal investigations.

  The government is encouraging people to improve themselves by implementing a face recognition system for digital payments as a form of authentication.

- **Ethical**

  The product can be regarded ethical as long as it does not infringe the privacy of individuals or disrupt public safety, is stored safely, and is not vulnerable to bad actors.

- **Health and Safety**

  The face recognition system wouldn’t affect someone’s health and is it safe for every human being in terms of physical health.

- **Sustainability**

  As we can see, most devices, such as phones and now Windows 10, have implemented Face unlock features, which are becoming more popular by the day. As a result, we can predict that this technology will continue to develop and sustain itself in the future.

- **Inspectability**

  If face recognition technology falls into the wrong hands, it can cause a slew of concerns with inspectability. When this technology is used in a harmful way, it can invade people's privacy and destroy society's ethical ideals.

4.2.3. System Requirements

4.2.3.1. H/W Requirements (details about Application Specific Hardware)

CPU	Intel i5 8th generaation
Random Access Memory (RAM) required:	8 GB
RAM occupied:	1.38 GB on Google Colab
Storage Space:	40 GB of space on Google Colab

To implement this project work we do require a laptop with a high-definition camera or an external Webcam so that it could capture high quality images for training purposes.

We also need a fast GPU (Graphic Processing Unit) which is used while training the model of images, else the computation will be

continued on CPU which could take hours to compute with a huge dataset. That is the reason the model training is done on Google Colab which provides us GPU power for model training purposes. These were the main requirements of the project.

4.2.3.2. S/W Requirements (details about Application Specific Software)

OS required:	Windows 10/Mac/Linux
Base Programming Language:	Python 3.6
Libraries / Framework:	Tensorflow, Keras, Opencv, Matplotlib
Integrated Development Environment (IDE):	Google Colab, Anaconda IDE

5. Results and Discussion

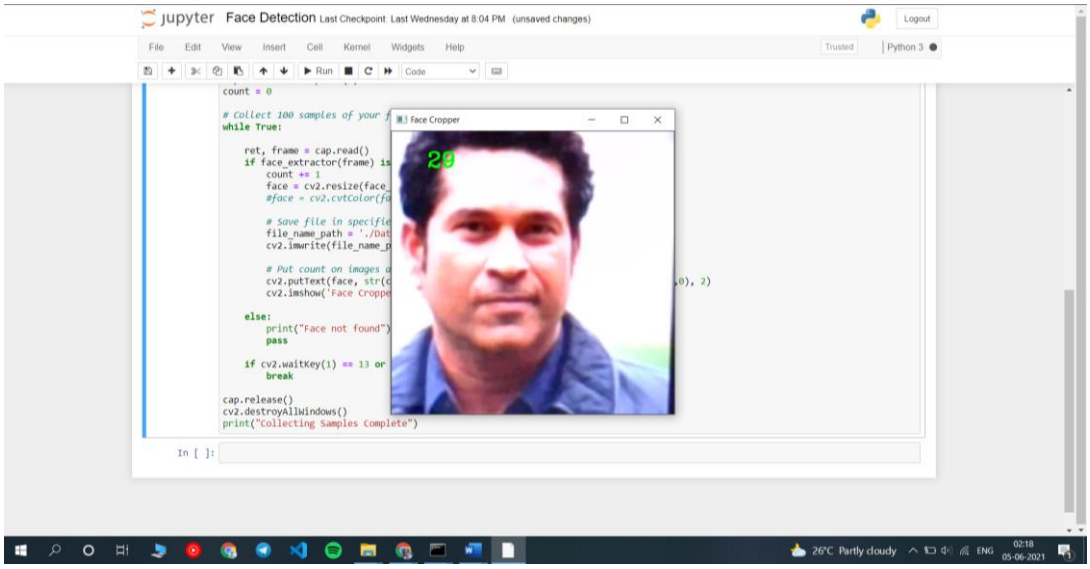


Figure 5.1 Face Detection

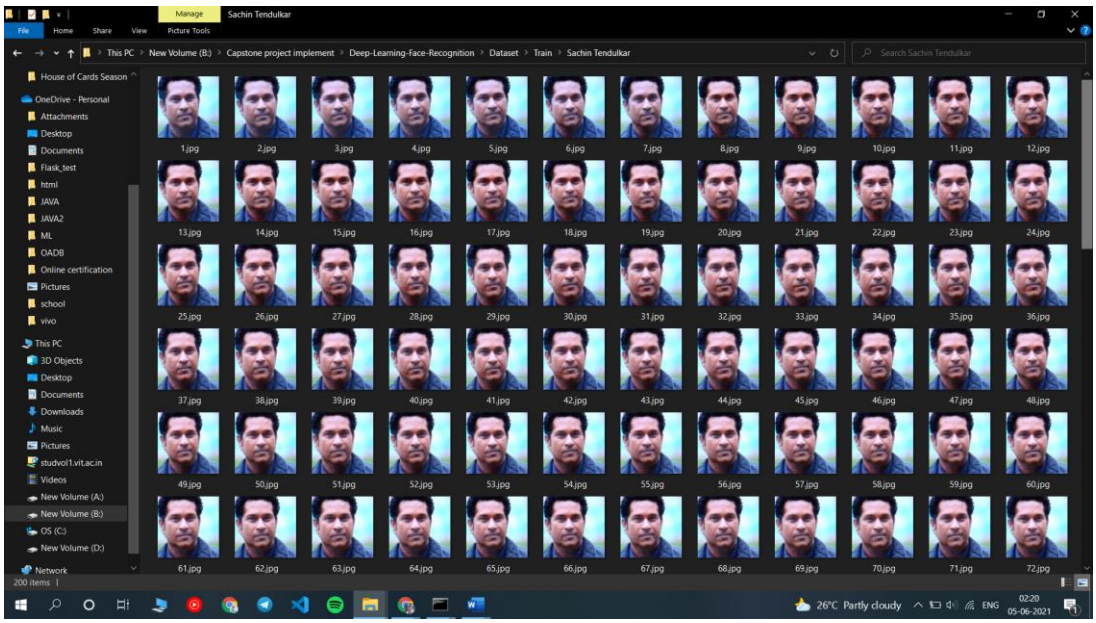


Figure 5.2 Dataset Collection

As we can see our face detector is successfully able to detect a face from the whole view visible to the web camera, moreover the counter of frames also seems to be perfect which counts how many images have been captured.

Like this we have captured a total of 2100 images, 1400 for

training and 700 for testing dataset.

Model Training

```
r = model.fit_generator(
  training_set,
  validation_data=test_set,
  epochs=10,
  steps_per_epoch=len(training_set),
  validation_steps=len(test_set),
)
```

```
/usr/local/lib/python3.7/dist-packages/tensorflow/python/keras/engine/training.py:1940: UserWarning: `Model.fit_generator` is deprecated and will
  warnings.warn('`Model.fit_generator` is deprecated and '
Epoch 1/10
88/88 [==============================] - 340s 4s/step - loss: 0.2182 - accuracy: 0.9614 - val_loss: 3.6632e-05 - val_accuracy: 1.0000
Epoch 2/10
88/88 [==============================] - 282s 3s/step - loss: 2.1711e-06 - accuracy: 1.0000 - val_loss: 3.8048e-05 - val_accuracy: 1.0000
Epoch 3/10
88/88 [==============================] - 277s 3s/step - loss: 4.2138e-04 - accuracy: 1.0000 - val_loss: 0.0230 - val_accuracy: 0.9914
Epoch 4/10
88/88 [==============================] - 283s 3s/step - loss: 1.0720e-04 - accuracy: 1.0000 - val_loss: 3.7712e-05 - val_accuracy: 1.0000
Epoch 5/10
88/88 [==============================] - 280s 3s/step - loss: 1.8110e-06 - accuracy: 1.0000 - val_loss: 3.2931e-05 - val_accuracy: 1.0000
Epoch 6/10
88/88 [==============================] - 284s 3s/step - loss: 4.8128e-06 - accuracy: 1.0000 - val_loss: 2.9951e-05 - val_accuracy: 1.0000
Epoch 7/10
88/88 [==============================] - 292s 3s/step - loss: 4.4254e-05 - accuracy: 1.0000 - val_loss: 2.0884e-06 - val_accuracy: 1.0000
Epoch 8/10
88/88 [==============================] - 296s 3s/step - loss: 7.3102e-07 - accuracy: 1.0000 - val_loss: 1.8883e-06 - val_accuracy: 1.0000
Epoch 9/10
88/88 [==============================] - 296s 3s/step - loss: 0.0029 - accuracy: 0.9986 - val_loss: 0.0016 - val_accuracy: 1.0000
Epoch 10/10
88/88 [==============================] - 294s 3s/step - loss: 0.0180 - accuracy: 0.9986 - val_loss: 8.7692e-05 - val_accuracy: 1.0000
```

Figure 5.3 Model Training and Prediction

As we can see we are able to get training accuracy on average about 99% and validation accuracy average 99%. Even the training and validation loss during the model training has appeared to be very less which is quite good for the model generation.

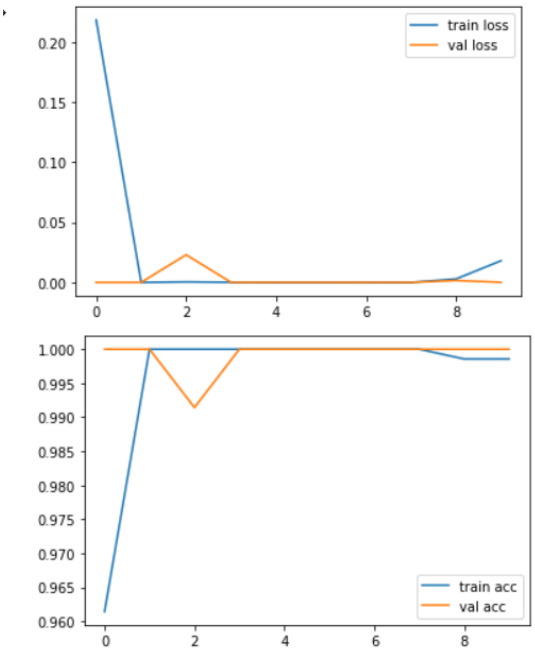


Figure 5.4 Graph presenting Train loss and Validation loss, Train accuracy and Validation accuracy

6. References

References

Weblinks:

1. Papageorgiou, C.P. & Oren, Michael & Poggio, Tomaso. (1998). General framework for object detection. Proceedings of the IEEE International Conference on Computer Vision. 6:. 555 - 562. 10.1109/ICCV.1998.710772.

2. KH Teoh et al 2021 J. Phys.: Conf. Ser. 1755 012006

3. Coşkun, Musab & Uçar, Ayşegül & yıldırım, Özal & Demir, Yakup. (2017). Face Recognition Based on Convolutional Neural Network.. 10.1109/MEES.2017.8248937.

4. Hassan, Rondik & Mohsin Abdulazeez, Adnan. (2021). Deep Learning Convolutional Neural Network for Face Recognition: A Review. 10.5281/zenodo.4471013.

5. Image-based Face Detection and Recognition: “State of the Art” Faizan Ahmad , Aaima Najam and Zeeshan Ahmed.

6. Cuimei, Li & Zhiliang, Qi & Nan, Jia & Jianhua, Wu. (2017). Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. 483-487. 10.1109/ICEMI.2017.8265863.

7. Samiksha Malhotra et al 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1022 012045

8. Illumination-robust face recognition based on deep convolutional neural networks architectures Ridha Ilyas Bendjillali1 , Mohammed Beladgham2 , Khaled Merit3 , Abdelmalik Taleb-Ahmed4

9. Hongling Chen and Chen Haoyu 2019 J. Phys.: Conf. Ser. 1229 012015

10. Almabdy, Soad; Elrefaei, Lamiaa. 2019. "Deep Convolutional Neural Network-Based Approaches for Face Recognition" *Appl. Sci.* 9, no. 20: 4397. https://doi.org/10.3390/app9204397

APPENDIX A

After going through the report, we have understood the methodology of Face recognition algorithms and their respective implementations. We have learned how InceptionResNetV2 , a transfer learning technique is able to able to generate a model to predict faces using its pre trained layers on image net dataset.
Here is a code that shows us how a model is generated using InceptionResNetV2

FUTURE SCOPE

If we talk about the future of the project, face recognition can be enhanced using an automated Application which will have a much user friendly UI so that the user could feed his face image into the database using the interface instead of running the codes.

SOME DRAWBACKS

There is a drawback here, we create a dataset using face detection and generate the model out of it, but if we have to add more human faces in the dataset, we would have to train the model again, like any other machine learning model as the dataset has changed and model won't be able to predict the new individual in the dataset which takes some time depending on the computer to computer, but this process is slow.