A Project Report on

Facial Recognition based Attendance System

Submitted in partial fulfillment of the requirements for the award
of the degree of

Bachelor of Engineering

in

Computer Engineering

by

Yash Manjardekar(15102013)
Sathyaveer Karmarkar(15102031)
Vasan Naddoni(15102015)
Vishal Chavan(15102033)

Under the Guidance of

Prof. Sachin H. Malave

Department of Branch Name
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2018-2019

Approval Sheet

This Project Report entitled ***"Facial Recognition based Attendance System"*** Submitted by ***"Yash Manjardekar"(15102013), "Sathyaveer Karmarkar"(15102031), "Vasan Naddoni"(15102015), "Vishal Chavan"(15102033)*** is approved for the partial fulfillment of the requirement for the award of the degree of ***Bachelor of Engineering*** in ***Computer Engineering*** from ***University of Mumbai***.

Examiners

1.

2.

Place : A. P. Shah Institute of Technology, Thane

Date :

CERTIFICATE

This is to certify that the project entitled ***"Facial Recognition Based Attendance System"*** submitted by ***"Yash Manjardekar" (15102013), "Sathyaveer Karmarkar" (15102031), "Vasan Naddoni" (15102015), "Vishal Chavan" (1512033)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering*** in ***Computer Engineering***, to the University of Mumbai,is a bonafide work carried out during academic year 2018-2019.

(Prof. Sachin H. Malave)
Guide

(Name and Sign)
External Examiner

(Prof. Amol R. Kalugade)
Project Co-ordinator

(Prof. Sachin H. Malave)
Head of Department

Place: A. P. Shah Institute of Technology, Thane
Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

_____

(Signature)

_____

(Yash Manjardekar 15102013)
(Sathyaveer Karmarkar 15102031)
(Vasan Naddoni 15102015)
(Vishal Chavan 15102033)

Date:

Abstract

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts.

Automatic face recognition (AFR) technologies have seen dramatic improvements in performance over the past years, and such systems are now widely used for security and commercial applications. An automated system for human face recognition in a real time background for a college to mark the attendance of their employees. So Smart Attendance using Real Time Face Recognition is a real world solution which comes with day to day activities of handling employees. The task is very difficult as the real time background subtraction in an image is still a challenge (6). To detect real time human face are used and a simple fast Principal Component Analysis has used to recognize the faces detected with a high accuracy rate. The matched face is used to mark attendance of the employee.Our system maintains the attendance records of employees automatically. Manual entering of attendance in logbooks becomes a difficult task and it also wastes the time. So we designed an efficient module that comprises of face recognition to manage the attendance records of employees. Our module enrols the staffs face (3). This enrolling is a onetime process and their face will be stored in the database. During enrolling of face we require a system since it is a onetime process. You can have your own roll number as your employee id which will be unique for each employee. The presence of each employee will be updated in a database. The results showed improved performance over manual attendance management system. Attendance is marked after employee identification. This product gives much more solutions with accurate results in user interactive manner rather than existing attendance and leave management systems.

Key Words: Biometric, Recognition system, Linux server, Face Recognition, Deep Learning, Python.

Contents

List of Figures

List of Abbreviations

SVM:	Support Vector Machines
LDA:	Linear Discriminant Analysis
PCA:	Principal Component Analysis
ReLU:	Rectified Linear Unit
ANN:	Artificial Neural Network
CNN:	Convolutional Neural Network
BPNN:	Back Propogation Neural Network
FNN:	Feedforward Neural Networ
RNN:	Recurrent Neural Network
MNN:	Modular Neural Network

Chapter 1

Introduction

Face recognition is the task of identifying an already detected object as a known or unknown face.Often the problem of face recognition is confused with the problem of face detectionFace Recognition on the other hand is to decide if the "face" is someone known, or unknown, using for this purpose a database of faces in order to validate this input face.

Maintaining the attendance is very important in all the institutes for checking the performance of employees. Every institute has its own method in this regard. Some are taking attendance manually using the old paper or file based approach and some have adopted methods of automatic attendance using some biometric techniques. But in these methods employees have to wait for long time in making a queue at time they enter the office. Many biometric systems are available but the key authentications are same is all the techniques. Every biometric system consists of enrolment process in which unique features of a person is stored in the database and then there are processes of identification and verification. These two processes compare the biometric feature of a person with previously stored template captured at the time of enrollment. Biometric templates can be of many types like Fingerprints, Eye Iris, Face, Hand Geometry, Signature, Gait and voice.Our system uses the face recognition approach for the automatic attendance of employees in the class room environment without students intervention. Face recognition consists of two steps, in first step faces are detected in the image and then these detected faces are compared with the database for verification.

A number of methods have been proposed for face detection i.e. Ada Boost algorithm, the Float Boost algorithm, the S-Ada Boost algorithm Support Vector Machines (SVM), and the Bayes classifier. The efficiency of face recognition algorithm can be increased with the fast face detection algorithm. Our system utilized this algorithm for the detection of faces in the class room image. Face recognition techniques can be Divided into two types Appearance based which use texture features that is applied to whole face or some specific Regions, other is Feature based which uses geometric features like mouth, nose, eyes, eye brows, cheeks and Relation between them. Statistical tools such as Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Kernel Methods, and Neural Networks, Eigen-faces have been used for construction of face templates.

1.1 Face Recognization:

DIFFERENT APPROACHES OF FACE RECOGNITION:

There are two predominant approaches to the face recognition problem: Geometric (feature based) and photometric (view based). As researcher interest in face recognition continued, many different algorithms were developed, three of which have been well studied in face recognition literature.

Recognition algorithms can be divided into two main approaches:

1. Geometric: Is based on geometrical relationship between facial landmarks, or in other words the spatial configuration of facial features. That means that the main geometrical features of the face such as the eyes, nose and mouth are first located and then faces are classified on the basis of various geometrical distances and angles between features.

2. Photometric stereo: Used to recover the shape of an object from a number of images taken under different lighting conditions. The shape of the recovered object is defined by a gradient map, which is made up of an array of surface normals (Zhao and Chellappa, 2006)

Popular recognition algorithms include:
1. Principal Component Analysis using Eigenfaces (PCA)
2. Linear Discriminate Analysis.
3. Elastic Bunch Graph Matching using the Fisherface algorithm.

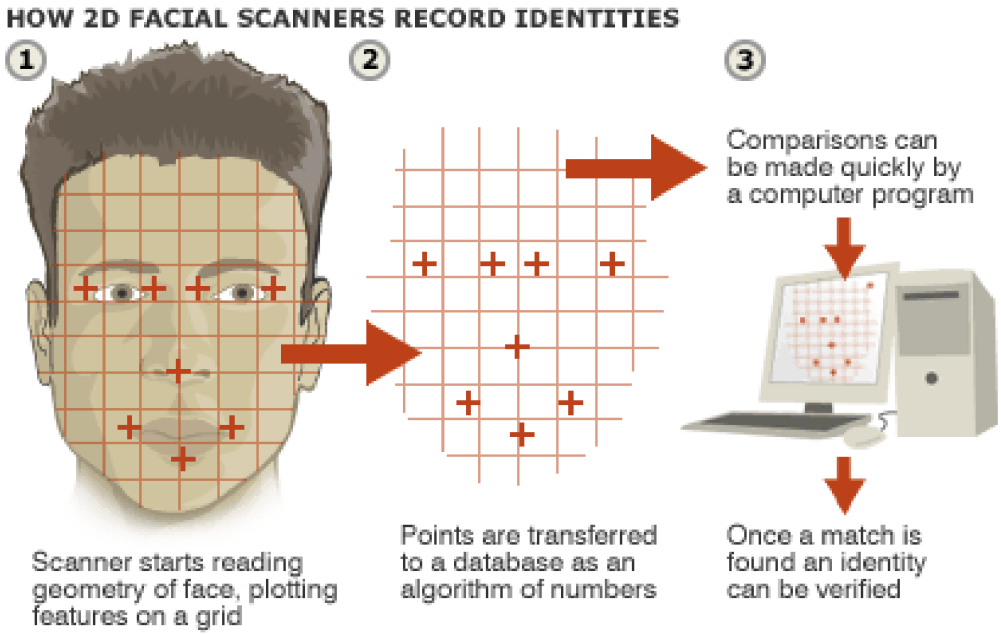

Figure 1.1: Basic Facial Recognition Process

1.2 Face Detection:

Face detection involves separating image windows into two classes; one containing faces tarning the background (clutter). It is difficult because although commonalities exist between faces, they can vary considerably in terms of age, skin colour and facial expression. The problem is further complicated by differing lighting conditions, image qualities and geometries, as well as the possibility of partial occlusion and disguise. An ideal face detector would therefore be able to detect the presence of any face under any set of lighting conditions.

The face detection system can be divided into the following steps:-

1. Pre-Processing: To reduce the variability in the faces, the images are processed before they are fed into the network. All positive examples that is the face images are obtained by cropping images with frontal faces to include only the front view. All the cropped images are then corrected for lighting through standard algorithms.

2. Recognition: Neural networks are implemented to recognize the image whether it belongs to a particular person of that class or not. Different network configurations are experimented with to optimize the results.

1.3 Project Objectives

This system will scan the face using camera and match the scanned with all entries in the database. Then the attendance of the particular entry will be marked if match found. Else it will display not recognized. Trying to nd a face within a large database of faces. In this approach the system returns a possible list of faces from the database. The most useful applications contain crowd surveillance, video content indexing, personal identication. Real time face recognition: Here, face recognition is used to identify a person on the spot and grant access to a building or a compound, thus avoiding security hassles. In this case the face is compared against a multiple training samples of a person

1.4 Project History

Scientists developed Facial Recognition Software in the 1960s. The scientists names were Woody Bledsoe, Helen Chan Wolf, and Charles Bisson. (There are two surprises to me here. First that scientists were working on this more than a half-century ago. Secondly, one of the scientists in the 1960s! was a woman.) Their programs required the administrator to locate features such as the eyes, ears, nose, and mouth on the photograph. It then calculated distances and ratios to a common reference point which was then compared to reference data. Certainly, the technology continued to advance decade by decade. In 1993 one of the US Defense agencies took it over. They named the project FERET or Face Recognition Technology Evaluation. In 2006, The Face Recognition Grand Challenge (FRGC) evaluated the latest face recognition algorithms available. High-resolution face images, 3D face scans, and iris images were used in the tests. Some of the algorithms were able to outperform human participants in recognizing faces and could uniquely identify identical twins.

A key factor of improving the quality of education is having students attend classes regularly. Traditionally students are stimulated to attend classes using attendance points which at the end of a semester constitute a part of a students final grade. However, traditionally this presents additional effort from the teacher, who must make sure to correctly mark attending students, which at the same time wastes a considerable amount of time from the teaching process. Furthermore it can get much more complicated if one has to deal with large groups of students. Maintaining the attendance is very important and compulsory in all the institutes for checking the performance of students. Every institute has its own method in this regard. Some are taking attendance manually using the old paper or file based approach and some have adopted methods of automatic attendance using some biometric techniques. There are many automatic methods available for this purpose i.e. biometric attendance. All these methods also waste time because students have to make a queue to touch their thumb on the scanning device. Organizations of all sizes use attendance systems to record when student or employees start and stop work, and the department where the work is performed. When it comes to schools and universities, the attendance monitoring system is a great help for parents and teachers both. Parents are never uninformed of the dependability of their children in the class if the university is using an attendance monitoring system

1.5 Neural Networks

The term "Neural networks" is a very evocative one. It suggests machines that are something like brains and is potentially laden with the science fiction connotations of the Frankenstein mythos. One of the main tasks of this book is to demystify neural networks and show how, while they indeed have something to do with brains, their study also makes contact with other branches of science, engineering and mathematics. The aim is to do this in as non-technical a way as possible, although some mathematical notation is essential for specifying certain rules, procedures and structures quantitatively. Nevertheless, all symbols and expressions will be explained as they arise so that, hopefully, these should not get in the way of the essentials: that is, concepts and ideas that may be described in words.

1.6 What are neurons?

Neurons are the part of Artificial Neural Networks in Machine Learning which is inspired by brain neural system. Our brain contains billions of connected neurons forming a neural network. Each neuron in this network has a basic structure. Each neuron receives inputs from other neurons through dendrites. Then there is some processing on the input and the calculated result is passed to other neurons connected through axon. Artificial Neural Network has the same structure too.Here each node represents a neuron. Each node is associated to an activation function which transforms the input to a output value. Some activation functions we use includes sigmoid , Tanh, ReLU. Output value of these functions then acts as input to next connected neurons. Final output helps in deciding the class of input data.

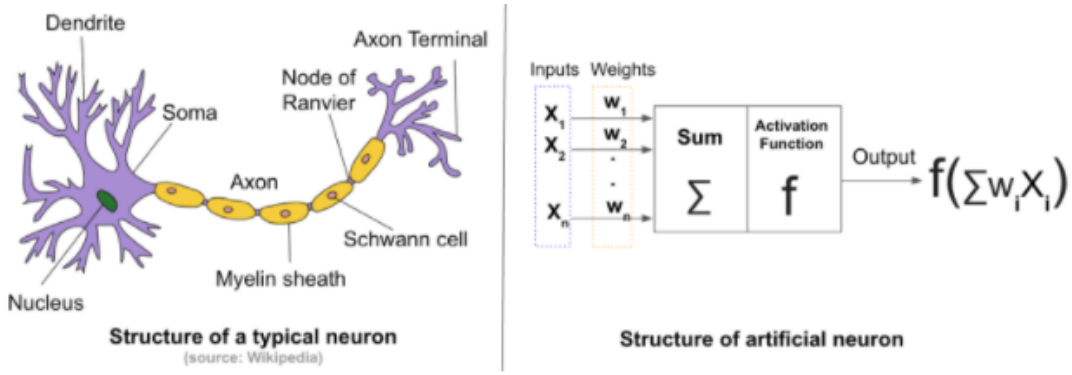

Figure 1.2: Difference between typical and artificial neuron

1.7 What are neural networks?

A neural network is an interconnected assembly of simple processing elements, units or nodes, whose functionality is loosely based on the animal neuron. The processing ability of the network is stored in the interunit connection strengths, or weights, obtained by a process of adaptation to, or learning from, a set of training patterns.

The term "network" will be used to refer to any system of artificial neurons. This may range from something as simple as a single node to a large collection of nodes in which each one is connected to every other node in the net. Each node is now shown by only a circle but weights are implicit on all connections. The nodes are arranged in a layered structure in which each signal emanates from an input and passes via two nodes before reaching an output beyond which it is no longer transformed. This feedforward structure is only one of several available and is typically used to place an input pattern into one of several classes according to the resulting pattern of outputs. For example, if the input consists of an encoding of the patterns of light and dark in an image of handwritten letters, the output layer (topmost in the figure) may contain 26 nodes one for each letter of the alphabetto flag which letter class the input character is from. This would be done by allocating one output node per class and requiring that only one such node fires whenever a pattern of the corresponding class is supplied at the input.

1.8 Why study neural networks?

This question is pertinent here because, depending on one's motive, the study of connectionism can take place from differing perspectives. It also helps to know what questions we are trying to answer in order to avoid the kind of religious wars that sometimes break out when the words "connectionism" or "neural network" are mentioned.

Neural networks are often used for statistical analysis and data modelling, in which their role is perceived as an alternative to standard nonlinear regression or cluster analysis techniques (Cheng Titterington 1994). Thus, they are typically used in problems that may be couched in terms of classification, or forecasting. Some examples include image and speech recognition, textual character recognition, and domains of human expertise such as medical diagnosis, geological survey for oil, and financial market indicator prediction

Neuroscientists and psychologists are interested in nets as computational models of the animal brain developed by abstracting what are believed to be those properties of real nervous tissue that are essential for information processing. The artificial neurons that connectionist models use are often extremely simplified versions of their biological counterparts and many neuroscientists are sceptical about the ultimate power of these impoverished models, insisting that more detail is necessary to explain the brain's function. Only time will tell but, by drawing on knowledge about how real neurons are interconnected as local "circuits", substantial inroads have been made in modelling brain functionality. A good introduction to this programme of computational neuroscience is given by Churchland Sejnowski (1992).

1.9 What is Artificial Neural Networks

Artificial Neural Network (ANN) is an efficient computing system whose central theme is borrowed from the analogy of biological neural networks. ANNs are also named as artificial neural systems, or parallel distributed processing systems, or connectionist systems. ANN acquires a large collection of units that are interconnected in some pattern to allow communication between the units. These units, also referred to as nodes or neurons, are simple processors which operate in parallel.

Every neuron is connected with other neuron through a connection link. Each connection link is associated with a weight that has information about the input signal. This is the most useful information for neurons to solve a particular problem because the weight usually excites or inhibits the signal that is being communicated. Each neuron has an internal state, which is called an activation signal. Output signals, which are produced after combining the input signals and activation rule, may be sent to other units.

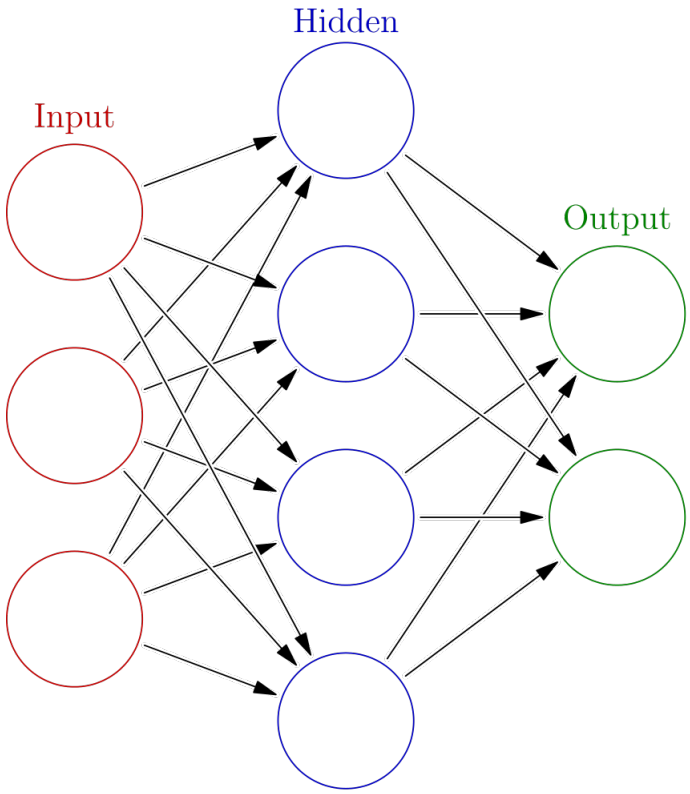

Figure 1.3: General structure of Artificial Neural Network.

1.10 Types of Artificial Neural Networks

Artificial neural networks are computational models which work similar to the functioning of a human nervous system. There are several kinds of artificial neural networks. These type of networks are implemented based on the mathematical operations and a set of parameters required to determine the output. Lets look at some of the neural networks:

1.10.1 Feedforward Neural Network Artificial Neuron

This neural network is one of the simplest form of ANN, where the data or the input travels in one direction. The data passes through the input nodes and exit on the output nodes. This neural network may or may not have the hidden layers. In simple words, it has a front propagated wave and no back propagation by using a classifying activation function usually.

Below is a Single layer feed forward network. Here, the sum of the products of inputs and weights are calculated and fed to the output. The output is considered if it is above a certain value i.e threshold(usually 0) and the neuron fires with an activated output (usually 1) and if it does not fire, the deactivated value is emitted (usually -1). Application of Feed forward neural networks are found in computer vision and speech recognition where classifying the target classes are complicated.

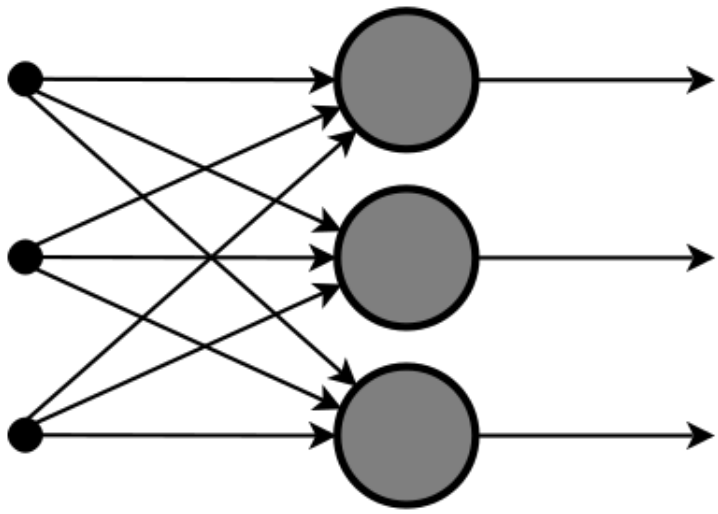

Figure 1.4: Feedforward Neural Network Artificial Neuron.

1.10.2 Radial basis function Neural Network

Radial basic functions consider the distance of a point with respect to the center. RBF functions have two layers, first where the features are combined with the Radial Basis Function in the inner layer and then the output of these features are taken into consideration while computing the same output in the next time-step which is basically a memory.

Below is a diagram which represents the distance calculating from the center to a point in the plane similar to a radius of the circle. Here, the distance measure used in euclidean, other distance measures can also be used. The model depends on the maximum reach or the radius of the circle in classifying the points into different categories. If the point is in or around the radius, the likelihood of the new point begin classified into that class is high. There can be a transition while changing from one region to another and this can be controlled by the beta function.

1.10.3 Kohonen Self Organizing Neural Network

The objective of a Kohonen map is to input vectors of arbitrary dimension to discrete map comprised of neurons. The map needs to me trained to create its own organization of the training data. It comprises of either one or two dimensions. When training the map the location of the neuron remains constant but the weights differ depending on the value. This self organization process has different parts, in the first phase every neuron value is initialized with a small weight and the input vector. In the second phase, the neuron closest to the point is the winning neuron and the neurons connected to the winning neuron will also move towards the point like in the graphic below. The distance between the point and the neurons is calculated by the euclidean distance, the neuron with the least distance wins. Through the iterations, all the points are clustered and each neuron represents each kind of cluster. This is the gist behind the organization of Kohonen Neural Network.

1.10.4 Recurrent Neural Network(RNN)

The Recurrent Neural Network works on the principle of saving the output of a layer and feeding this back to the input to help in predicting the outcome of the layer. Here, the first layer is formed similar to the feed forward neural network with the product of the sum of the weights and the features. The recurrent neural network process starts once this is computed, this means that from one time step to the next each neuron will remember some information it had in the previous time-step. This makes each neuron act like a memory cell in performing computations. In this process, we need to let the neural network to work on the front propagation and remember what information it needs for later use. Here, if the prediction is wrong we use the learning rate or error correction to make small changes so that it will gradually work towards making the right prediction during the back propagation. This is how a basic Recurrent Neural Network looks like,

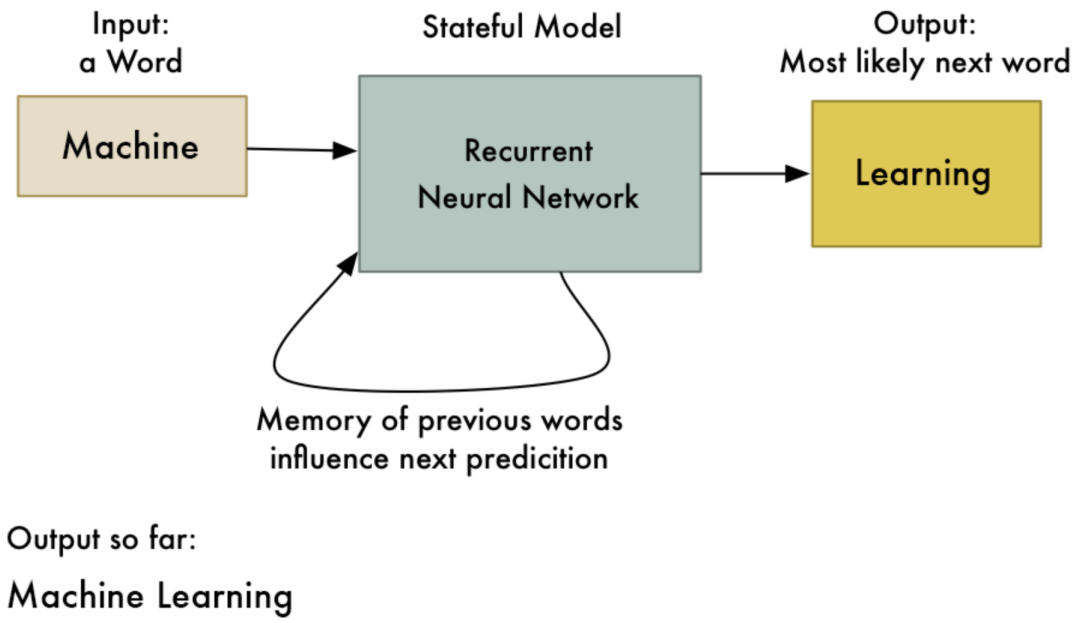

Figure 1.5: Recurrent Neural Network(RNN)

1.10.5 Convolutional Neural Network

Convolutional neural networks are similar to feed forward neural networks , where the neurons have learn-able weights and biases. Its application have been in signal and image processing which takes over OpenCV in field of computer vision.

Below is a representation of a ConvNet, in this neural network, the input features are taken in batch wise like a filter. This will help the network to remember the images in parts and can compute the operations. These computations involve conversion of the image from RGB or HSI scale to Gray-scale. Once we have this, the changes in the pixel value will help detecting the edges and images can be classified into different categories.
ConvNet are applied in techniques like signal processing and image classification techniques. Computer vision techniques are dominated by convolutional neural networks because of their accuracy in image classification. The technique of image analysis and recognition, where the agriculture and weather features are extracted from the open source satellites like LSAT to predict the future growth and yield of a particular land are being implemented.

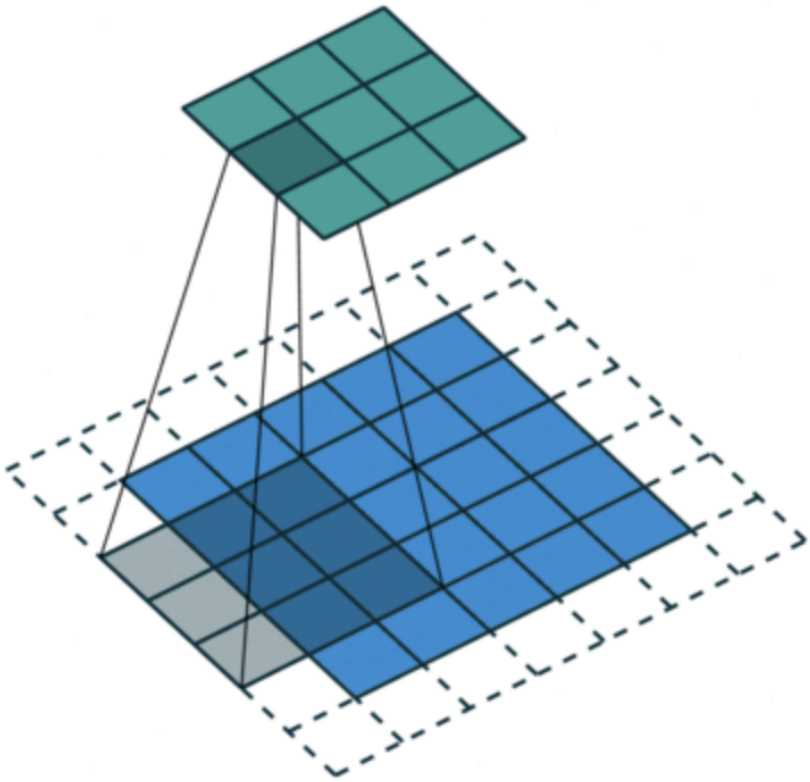

Figure 1.6: Convolutional Neural Network

1.10.6 Modular Neural Network

Modular Neural Networks have a collection of different networks working independently and contributing towards the output. Each neural network has a set of inputs which are unique compared to other networks constructing and performing sub-tasks. These networks do not interact or signal each other in accomplishing the tasks. The advantage of a modular neural network is that it breakdowns a large computational process into smaller components decreasing the complexity. This breakdown will help in decreasing the number of connections and negates the interaction of these network with each other, which in turn will increase the computation speed. However, the processing time will depend on the number of neurons and their involvement in computing the results.

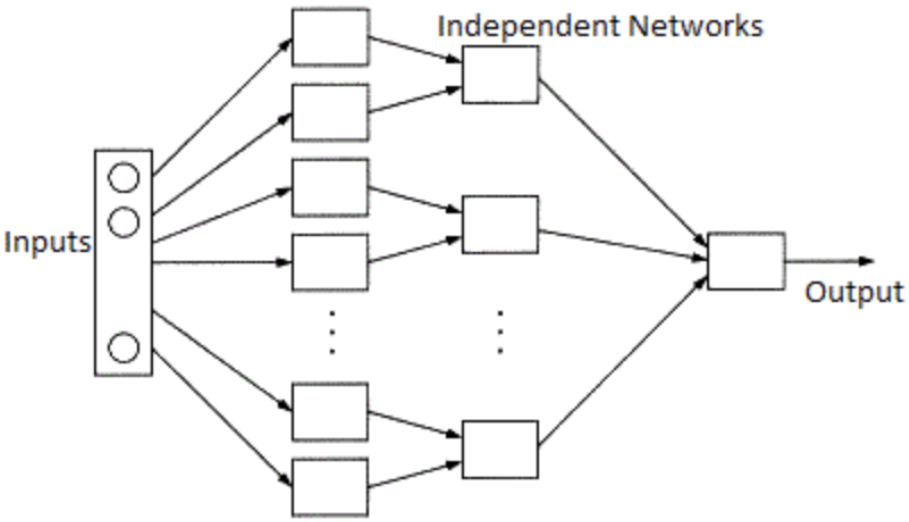

Figure 1.7: Modular Neural Network

1.11 Summary

Artificial neural networks may be thought of as simplified models of the networks of neurons that occur naturally in the animal brain. From the biological viewpoint the essential requirement for a neural network is that it should attempt to capture what we believe are the essential information processing features of the corresponding "real" network. For an engineer, this correspondence is not so important and the network offers an alternative form of parallel computing that might be more appropriate for solving the task in hand. The simplest artificial neuron is the threshold logic unit or TLU. Its basic operation is to perform a weighted sum of its inputs and then output a "1" if this sum exceeds a threshold, and a "0" otherwise. The TLU is supposed to model the basic "integrate-and-fire" mechanism of real neurons.

Chapter 2

Literature Review

Zuzana Bukovcikov, Dominik Sopiak, Milo Oravec, Jarmila Pavlovicov carried out the research on "Face VerificationUsing Convolutional Neural Networks with Siamese Architecture". The main objective of their study was CelebFaces Attributes Dataset (CelebA) is a large-scale face attributes dataset with more than 200 celebrity images, each with 40 attribute annotations. The images in this dataset cover large pose variations and background clutter. CelebA has large diversities, large quantities, and rich annotations, including 10,177 number of subjects, 202,599 number of face images, and 5 landmark locations, 40 binary attributes annotations per image. Number of images per subject is between 1 and 35. The implemented convolutional network, is based on commonly used AlexNet architecture, however we change the last fully connected layers as well as the size of kernels in convolutions. Two face images are inputs for our system and system should decide if there is the same person on both images or there are two different peoples. Images of the same person are called genuine pairs. Images of different persons are called impostor pairs. For pair classification we use Siamese architecture, which was proposed in and modified for face verification. Another option to increase success rate is improve- ment of face detection algorithm. Also there are many ways how to change architecture of CNN.[1]

Gregory Koch, Richard Zemel, Ruslan Salakhutdinov carried out research on "Siamese Neural Networks for One-shot Image Recognition".They described that the process of learning good features for machine learning applications can be very computationally expensive and may prove difficult in cases where little data is available. A prototypical example of this is the one-shot learning setting, in which we must correctly make predictions given only a single example of each new class. In this paper, we explore a method for learning siamese neural networks which employ a unique structure to naturally rank similarity between inputs. Once a network has been tuned, we can then capitalize on powerful discriminative features to generalize the predictive power of the network not just to new data, but to entirely new classes from unknown distributions. Using a convolutional architecture, we are able to achieve strong results which exceed those of other deep learning models with near state-of-the-art performance on one-shot classification tasks. Their research was that One-shot learning can be directly addressed by developing domain-specific features or inference procedures which possess highly discriminative properties for the target task. As a result, systems which incorporate these methods tend to excel at similar instances but fail to offer robust solutions that may be applied to other types of problems.[2]

Y. H. Kim, H. Kim, S. W. Kim, H. Y. Kim, and S. J. Ko carried out research on "Illumination normalization using convolutional neural network with application to face recognition". Their research was that Based on the commonly used assumption about the illumination field, the Illumination Elimination layers generate illumination insensitive ratio images by calculating the ratio between the output pairs produced from the Local Pattern Extraction layers.For training the proposed network, the results produced by the Weber fraction-based IN methods as ground truths are utilized. The experimental results demonstrate that the proposed network performs better in terms of Face Recognition accuracy compared with the conventional non-CNN-based method and it can be combined with any CNN-based face classifier. The proposed Illumination Normalization-Net realizes the kernel-based R-field estimation method by including the Local Pattern Extraction and Illumination Elimination layers. Furthermore, any ground truth images can be employed to train the proposed network which maps the illumination-variant face image to the final feature map. Our experimental results show that Illumination Normalization-Net achieves superior performance in terms of the FR accuracy on the extended Yale B and CMU-PIE databases.[3]

Yoshua Bengio carried out research on "Learning Deep Architectures for AI."Their research states that theoretical results strongly suggest that in order to learn the kind of complicated functions that can represent high-level abstractions (e.g. in vision, language, and other AI-level tasks), one needs deep architectures.Deep architectures are composed of multiple levels of non-linear operations, such as in neural network many hidden layers or in complicated propositional formulae re-using many sub-formulae. Searching the parameter space of deep architectures is a difficult optimization task, but learning algorithms such as those for Deep Belief Networks have recently been proposed to tackle this problem with notable success, beating the state-of-the-art in certain areas. This paper discusses the motivations and principles regarding learning algorithms for deep architectures, in particular those exploiting as building blocks unsupervised learning of single-layer models such as Restricted Boltzmann Machines, used to construct deeper models such as Deep Belief Networks.[4]

Yaniv Taigman, Ming Yang, MarcAurelio Ranzato, and Lior Wolf carried out research on "DeepFace: Closing the Gap to Human-Level Performance in Face Verification" Their research states that In modern face recognition, the conventional pipeline consists of four stages: detect align represent classify. We revisit both the alignment step and the representation step by employing explicit 3D face modeling in order to apply a piecewise affine transformation, and derive a face representation from a nine-layer deep neural network. This deep network involves more than 120 million parameters using several locally connected layers without weight sharing, rather than the standard convolutional layers. Trained it on the largest facial dataset to-date, an identity labeled dataset of four million facial images belonging to more than 4,000 identities. The learned representations coupling the accurate model-based alignment with the large facial database generalize remarkably well to faces in unconstrained environments, even with a simple classifier.[5]

Cheng, B. & D.M.Titterington carried out research on "Neural networks: a review from a statistical perspective". Their research states that the Statistical perspective on Artificial Neural Network which points out some of the links with statistical methodology and encourages cross-disciplinary research in the directions most likely to bear fruit. The area of statistical interest are briefly outlined and series of example indicates the flavour of ANN models. Various topics are treated in more depth and the neural network architecture and training rules are described providing a statistical commentary. The topics treated in this way are perceptrons Hopfield-type recurrent network and associative memory networks trained by so called unsupervised learning rules. Perceptrons are shown to have strong associations with discriminant analysis and regression, and unsupervised networks with cluster analysis. The paper concludes some thoughts on the future of the interface between neural networks and statistics.[6]

Churchland, P.S. & T.J.Sejnowski carried out a research on "The Computational Brain." They address the foundational ideas of the emerging field of computational neuroscience, examine a diverse range of neural network models, and consider future directions of the field. They discussed about the groups of neurons that interact to enable the organism to see, decide and move appropriately. They also examined the principles where by how networks of neurons represent and compute. Then they came to a conclusion that these are the central questions probed by the computational brain and now the computational brain is the first unified and broadly accessible book to bring together computational concepts and behavioural data within a neuro-biological framework.

Chapter 3

Siamese neural networks

Siamese neural network is a class of neural network architectures that contain two or more identical subnetworks. identical here means they have the same configuration with the same parameters and weights. Parameter updating is mirrored across both subnetworks. It is one of the prominent techniques of Convolutional Neural Network.

Siamese NNs are popular among tasks that involve finding similarity or a relationship between two comparable things. Some examples are paraphrase scoring, where the inputs are two sentences and the output is a score of how similar they are; or signature verification, where figure out whether two signatures are from the same person. Generally, in such tasks, two identical subnetworks are used to process the two inputs, and another module will take their outputs and produce the final output.

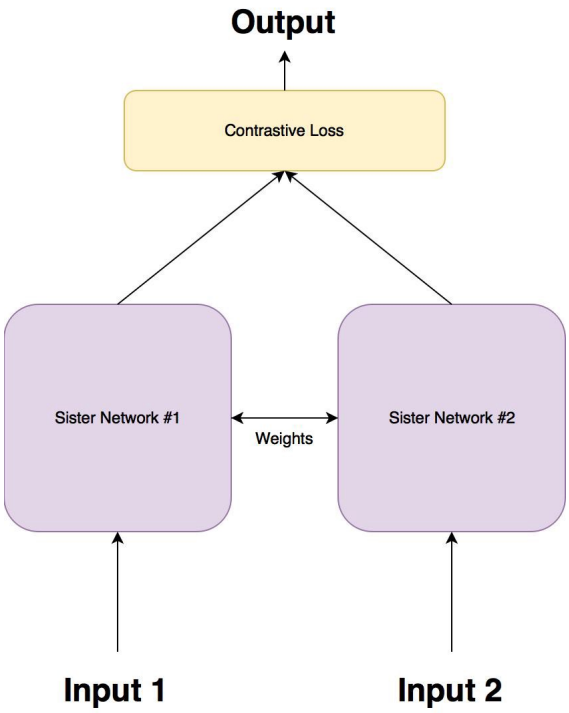

Figure 3.1: Siamese Neural Network

3.1 Architecture

In a Siamese network described in Figure 3.1, two inputs are taken and evaluated for a score. The two inputs are fed into Siamese convolutional networks that translate each image into latent encoding space. In this study, we vary these convolutional networks to determine which convolutional architecture produces the best encoding of handwriting. These latent encodings are then concatenated and used to produce class probabilities, or in this case the probabilities of whether or not two inputs are duplicates.

The convolutional neural network is trained such that each of the Siamese networks share weights, and thus each twin of the network outputs an encoding of an image using the same filters as the other. These networks are trained from scratch.

Although Siamese networks can be trained utilizing a distance function to maximize the distance between mismatches and minimize the distance between matches, this model requires search over multiple distance functions as well as different thresholds for distances. Furthermore, even if we were able to achieve good results with a distance measure, we remain unsure what the natural distance measure for the image space is. Early trials optimizing over the L2 Euclidean distance did not perform well on this dataset. To address this, we we basically calculate the difference between the tensors obtained by each input of convolution neural network.

The output of the Siamese networks for each image is encoded in a n-dimensional vector. Call the output of Siamese net 1 with image A,a, and the output of Siamese net 2, with image B, b.Then we basically calculate the difference between the tensors obtained by each input of convolution neural network.

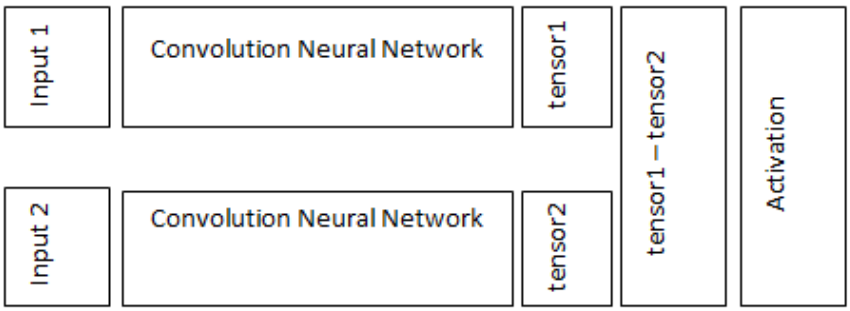

Figure 3.2: Overall Architecture of Siamese Neural Network

Siamese architectures are good in these tasks because:
1. Sharing weights across subnetworks means fewer parameters to train for, which in turn means less data required and less tendency to overfit.

2. Each subnetwork essentially produces a representation of its input. (Signature Feature Vector in the picture.) If your inputs are of the same kind, like matching two sentences

or matching two pictures, **it makes sense to use similar model to process similar inputs.** This way you have representation vectors with the same semantics, making them easier to compare.

3.2 Baseline Architecture

We designed our baseline intentionally to have few layers and large kernels, as a point of comparison to the later architectures which are deep with small kernels. The first convolutional layer uses 128 filters of size 3 x 3. The second convolutional layer uses 64 filters of size 3 x 3. The Third convolutional layer uses 32 filters of size 3 x 3.The Final convolutional layer uses 16 filters of size 3 x 3. Both are followed by a Relu non-linearity. We apply l2 regularization and dropout with a probability of 0.5.Then we a flatten layer followed by a dense layer of 16 units.

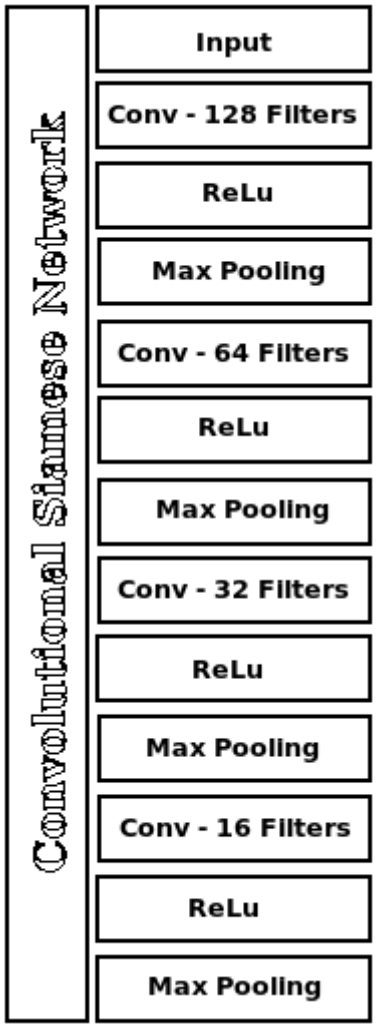

Figure 3.3: Baseline Network Architecture

3.3 Overview

Siamese nets were first introduced in the early 1990s by Bromley and LeCun to solve signature verification as an image matching problem. A Siamese neural network consists of twin networks which accept distinct inputs but are joined by an energy function at the top. This function computes some metric between the highest-level feature representation on each side. The parameters between the twin networks are tied. This strategy has two key properties: It ensures the consistency of its predictions. Weight tying guarantees that two extremely similar images could not possibly be mapped by their respective networks to very different locations in feature space because each network computes the same function. The network is symmetric: if we present two distinct images to the twin networks, the top conjoining layer will compute the same metric as if we were to we present the same two images but to the opposite twins. In LeCun et al., the authors used a contrastive energy function which contained dual terms to decrease the energy of like pairs and increase the energy of unlike pairs. However, in this paper we use the weighted L1 distance between the twin feature vectors h1 and h2 combined with a sigmoid activation, which maps onto the interval [0, 1]. Thus a cross-entropy objective is a natural choice for training the network. Note that in LeCun et al., they directly learned the similarity metric, which was implicitly defined by the energy loss, whereas we fix the metric as specified above, following the approach in Facebooks DeepFace paper. We now detail both the structure of the Siamese nets and the specifics of the learning algorithm used in our experiments.

Chapter 4

Methodology

4.1 Environment Setup

This includes the Operating System and all the softwares that we used for the project. The operating system used by us was macOS.Then we installed python 3 and pip. Later we updated the path for python to make it globally availaible. Then we installed various other modules of python using pip like "h5py, numpy, scikit-learn, sklearn, pyttsx3, datetime, pygame, opencv-python". The prominent module that we installed was keras which is used to implement neural networks in python easily. The backend we used for neural network was TensorFlow. The actual project backend used to update and store the attendance of students was non-relational database Mong Db. In this way the environment was setup.

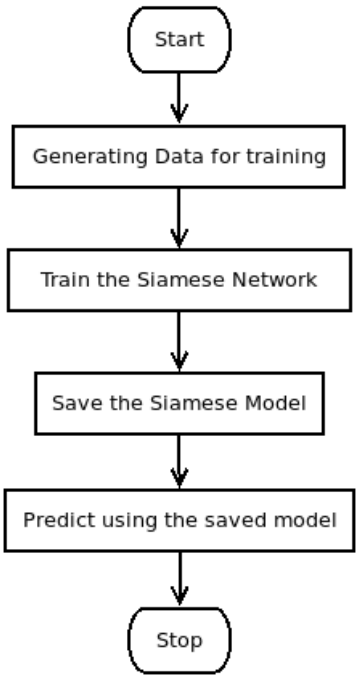

Figure 4.1: General flow

4.2 Getting Data for training

Here we are getting the images of people,either by capturing from the camera, or by saving the folder into people. When `"Generating_training_data.py"` this runs enter the name of the person followed by a index beginning from zero for example, if we want to generate data for "vasan", we will write "vasan0" and for the next name write "secondname1" and increment so on. Each image clicked form camera is saved as "1.jpg", the next as "2.jpg", in this way we take around 51 pictures which is then used for training. The code `"Generating_training_data.py"`, runs camera, which then captures image in various angles and various expressions, which inturn saves the image as stated above.

The actual mechanism behind the capturing of image is that, first after the camera is turned on using "cap=cv2.VideoCapture(0)" and then "ret,frame=cap.read()", we resize the image to fx=0.5 and fy=0.5. Then we flip the image, because it was taken from webcam and has to be flipped horizontally. In the next step, we convert the coloured instance of image to GRAY scale, which makes it easy for training the Neural Network. Now we crop the face of the person, by creating border around his face, so that the iamge becomes focused for Face Recgnition and remaining part gets cropped, so basically we use frame for cropping.

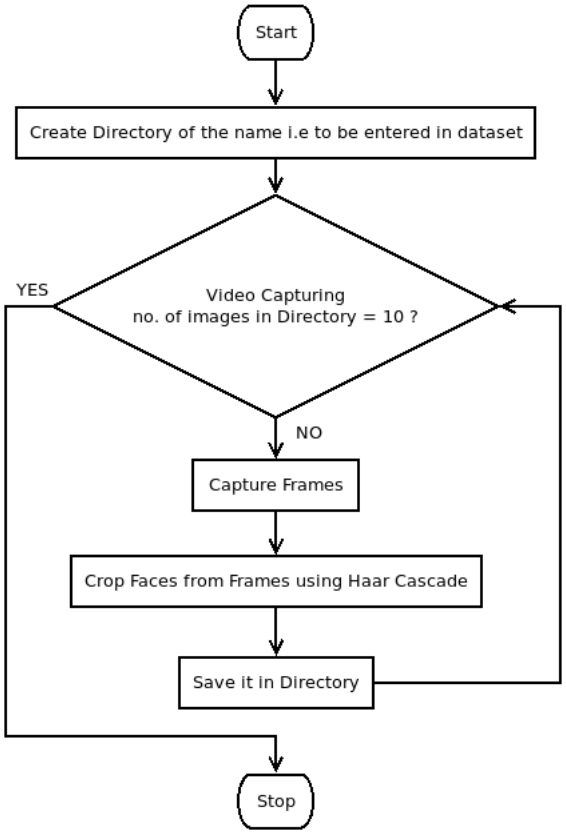

Figure 4.2: Getting Data for training

4.2.1 Dataset

Here we are using our own database of 60 college students in our class of BE Computer Department. We are basically taking at least 51 images of single student in various angles to test image recognition and train the neural network for multiple images in multiple angles.

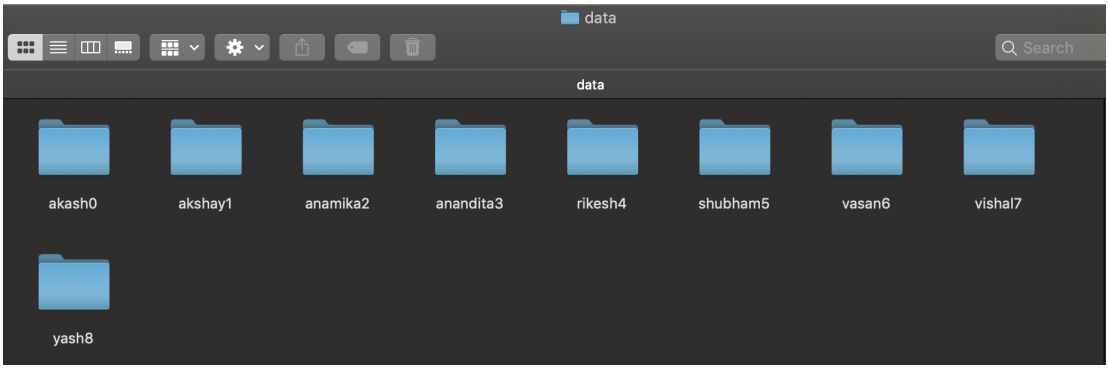

Figure 4.3: Dataset folders

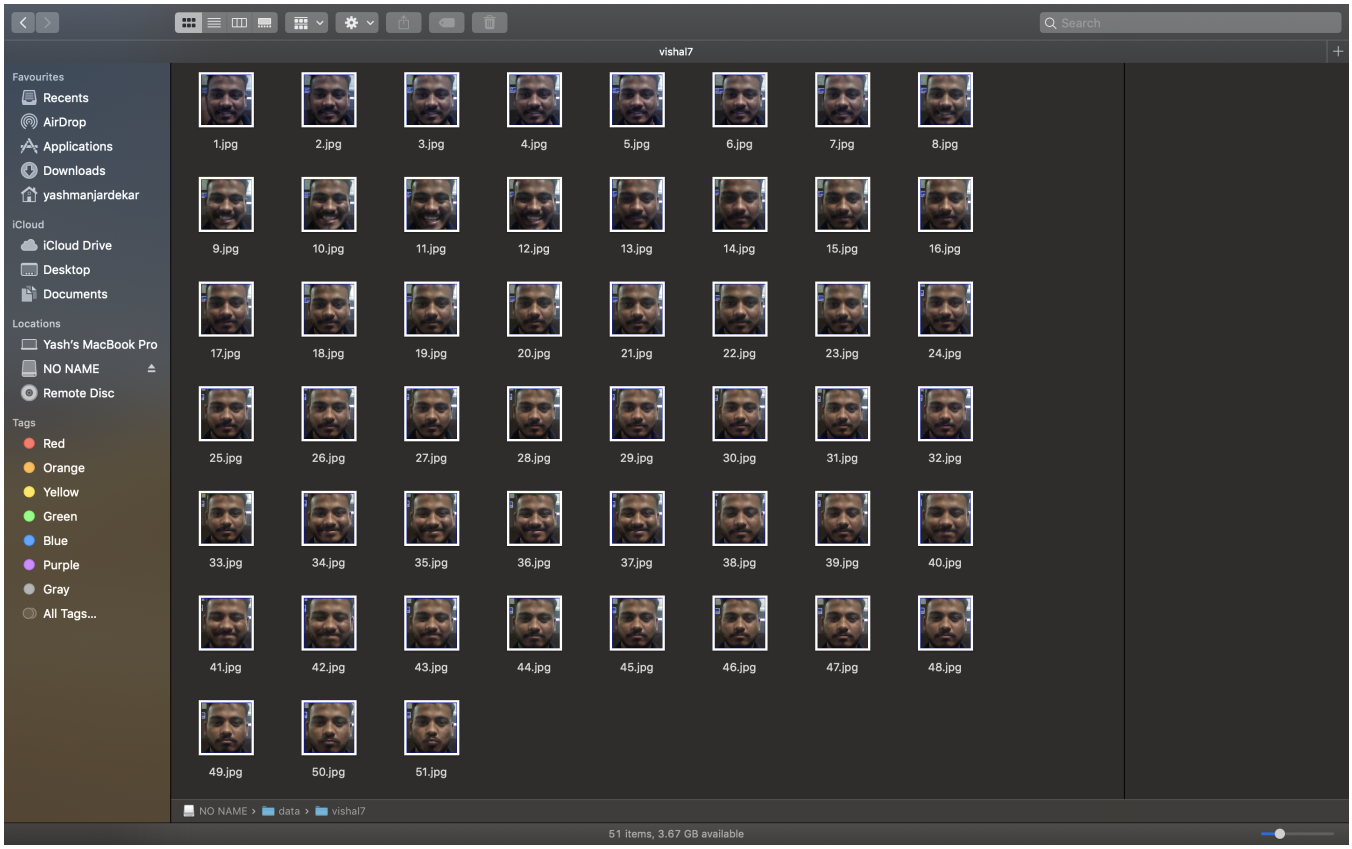

Figure 4.4: Pics inside the Dataset folder

4.3 Training the data

The actual training of the data happens here in this stage. In this, we are using siamese neural network, which is explained in Chapter 4. In this code, we are taking various libraries in python as input such as "numpy", "cv2", "keras", "os" and "matplotlib". Keras is the main library in python, which makes it easy for us to implement neural netwokrs.

Now in "trainer.py" change the number of classes according to number of people used for training in the people folder and then run trainer.py. The code "trainer.py" gets the path to our database to collect the data of images.Then it gets the image and resizes it to "160*160" scale. The value of each pixel of the image is converted into float with value between "0 to 1". Then later we expand the dimensions and append the image and its value to x and y array. The x array consists of image and y consists of its label.

Then we compare the two images. In this we actually compare each image with every other image in the dataset. With the help of labels we decide whether both the images are similar or different. For comparing we have created the two arrays which are `"left_input"` and `"right_input"` which are inputs to the siamese neurtal network and the targets array consists of the output, where the result is stored.

Now, in the next step, we implement various layers of convolutional neural network. It consists of max pooling layer, Conv2D layer and Flatten layer. Each layer does its own function. The maxpooling layer is used to find maximium of all the values for neurons in that layer. Whereas the Flatten layer Flattens the input but does not affect the batch size. The Conv2D layer is the main layer of convolutional neural network. There are various hidden layers available in keras library which can also be used for more efficiency.

While training, we implement the pairs of training and testing simultaneously. It works one after another, train first and test it and it goes on an on until all the epochs are processed.Then a ".Model" file is created, which we use for recogniton.The activation function that we use is sigmoid.

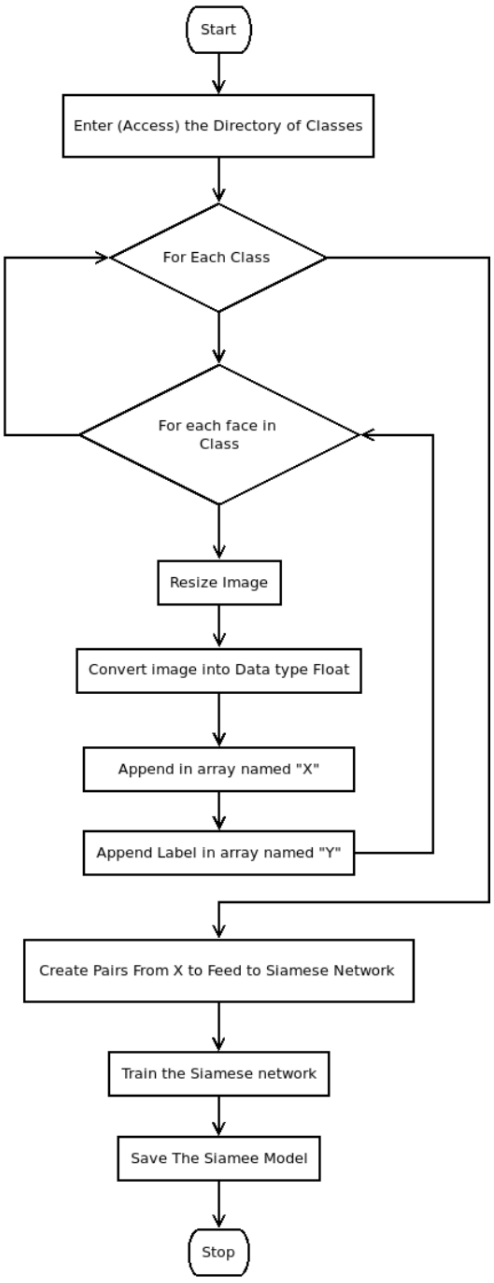

Figure 4.5: Training the model.

4.4 Recognition

In this, after we are training the data, we run "recognizer.py" to recognize the face and mark the attendance of the student. After execution of this code, the camera will get activated, and start capturing the image for recognizing the student. Initially, we import all the models and necessary things. Later, we import the ".Model" file that was created while training the data.

After importing all the files, we give the path to the people folder, which has original images of the students. Now, it tests for the the student in front of camera and searches for it in the database. For this purpose it again draws border for the persons face, similar to the one as it does while generating the training data. Now, it captures the input, resizes it to "160*160".Then it converts the value of each pixel equivalent to value between "0-1" as it does while training the dta. Further it again expands dimensions and appends the value in the array. Finally, it predicts both the images and gives output of whether or not the student is same and if detected, marks its attendance. In this way it will mark the attendance of each student as soon as it recognizes it.

Then this program exports a file in ".csv" format. Then we can use this file to check the details of attendance for each student.

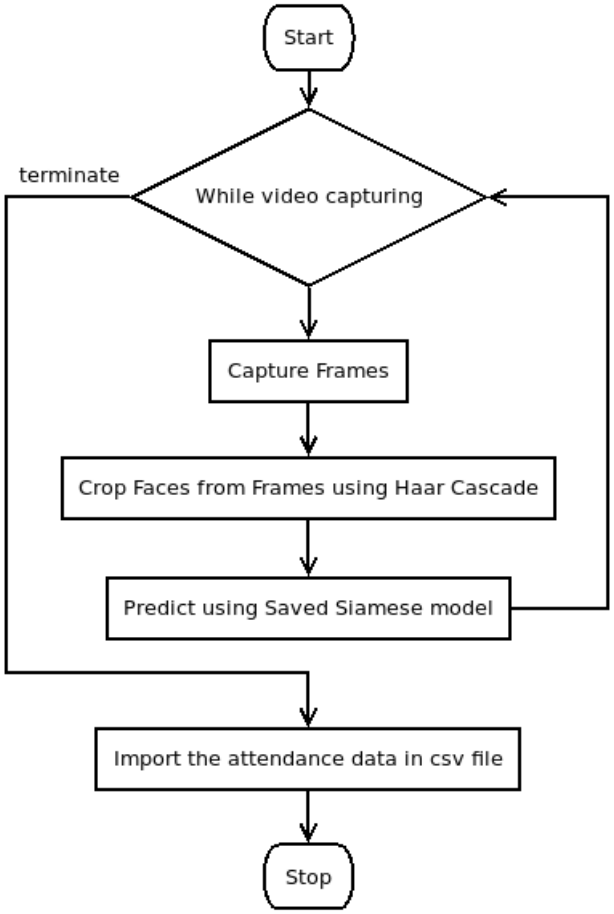

Figure 4.6: Recognition

4.5 Test Run

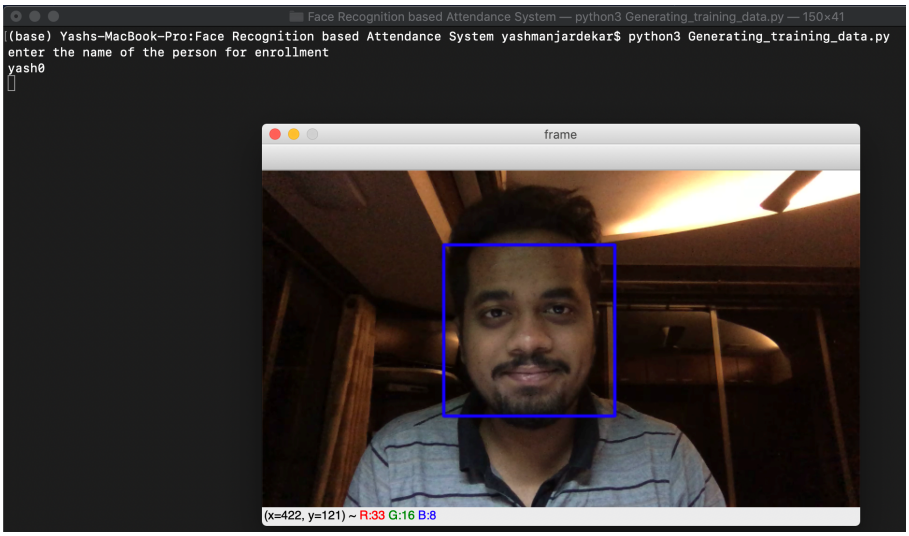

Figure 4.7: The camera will open to capture the images.

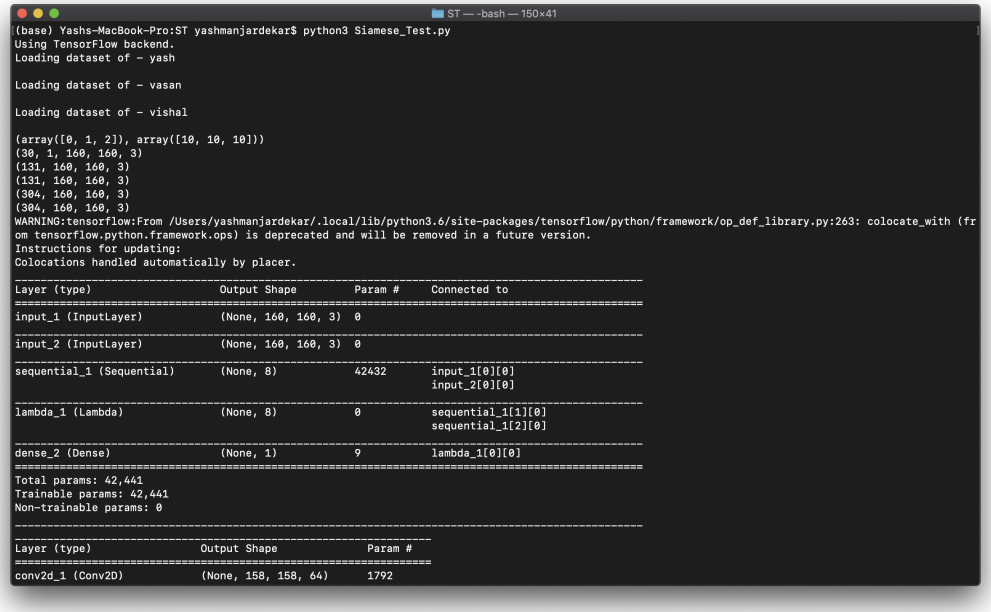

Figure 4.8: Now train the model by running the specific code.

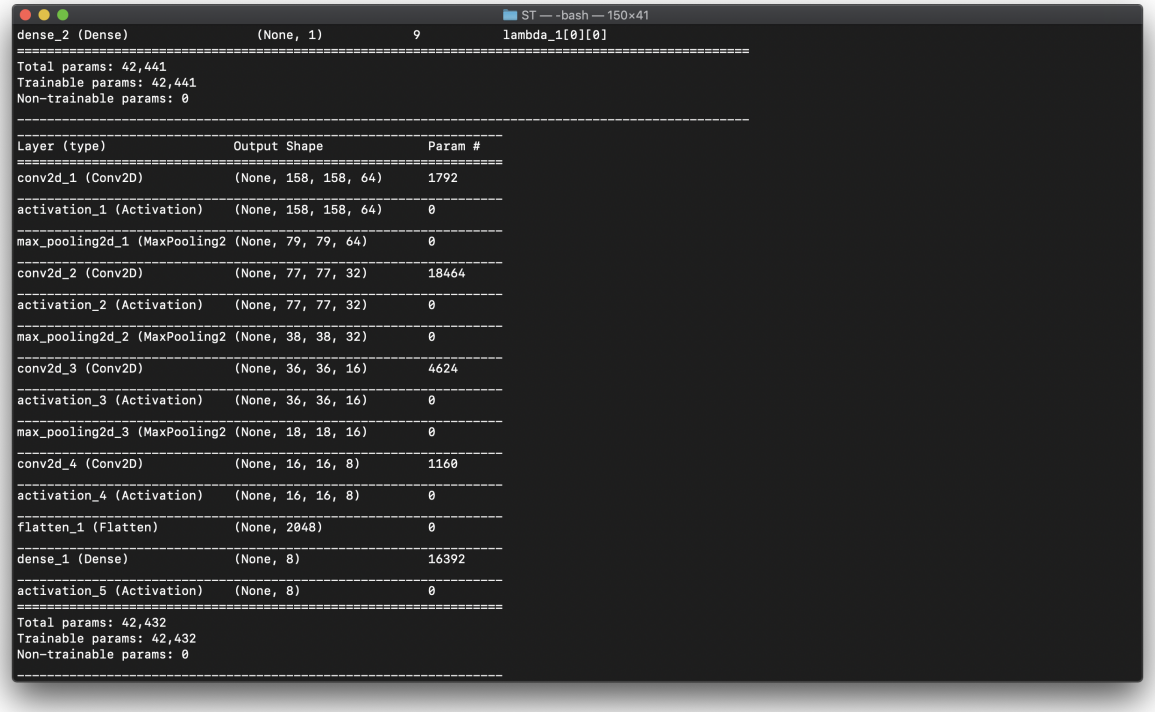

Figure 4.9: Model architure is shown.

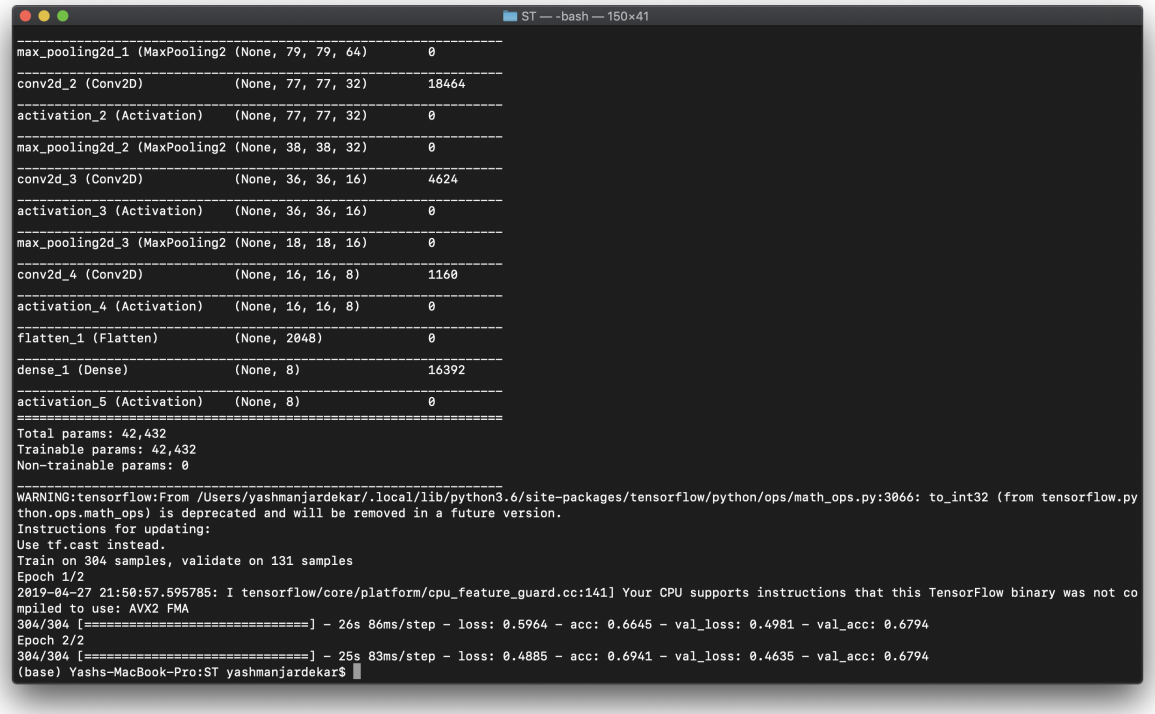

Figure 4.10: Epochs of the code is shown.

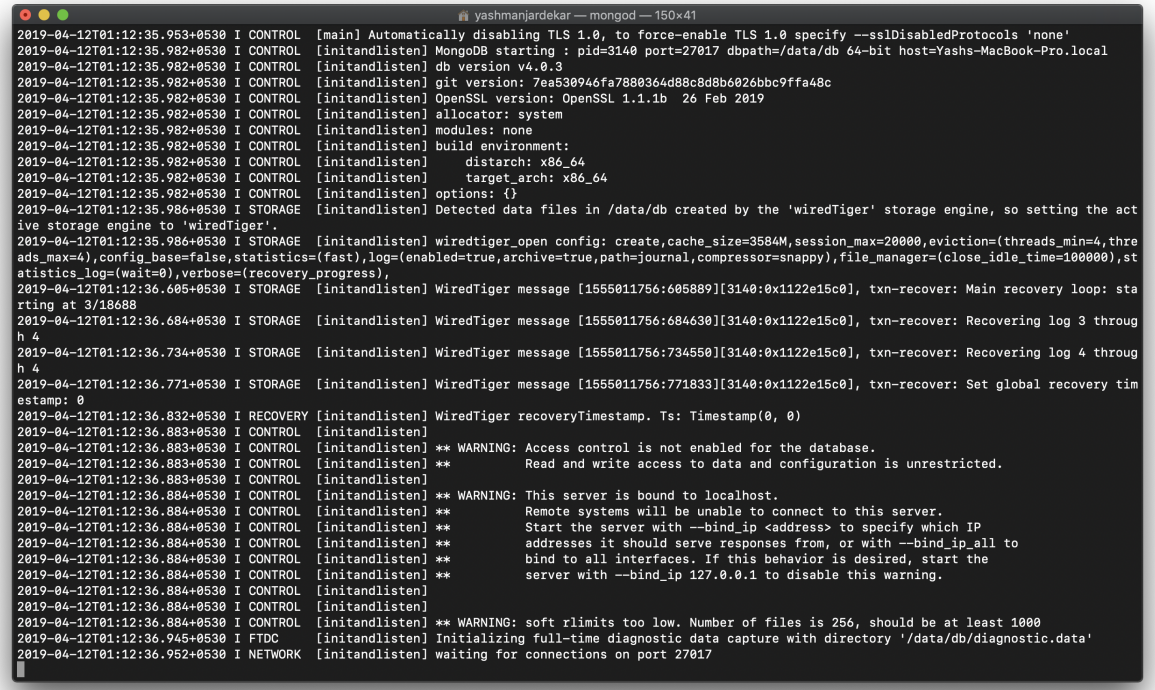

Figure 4.11: Starting mongod service

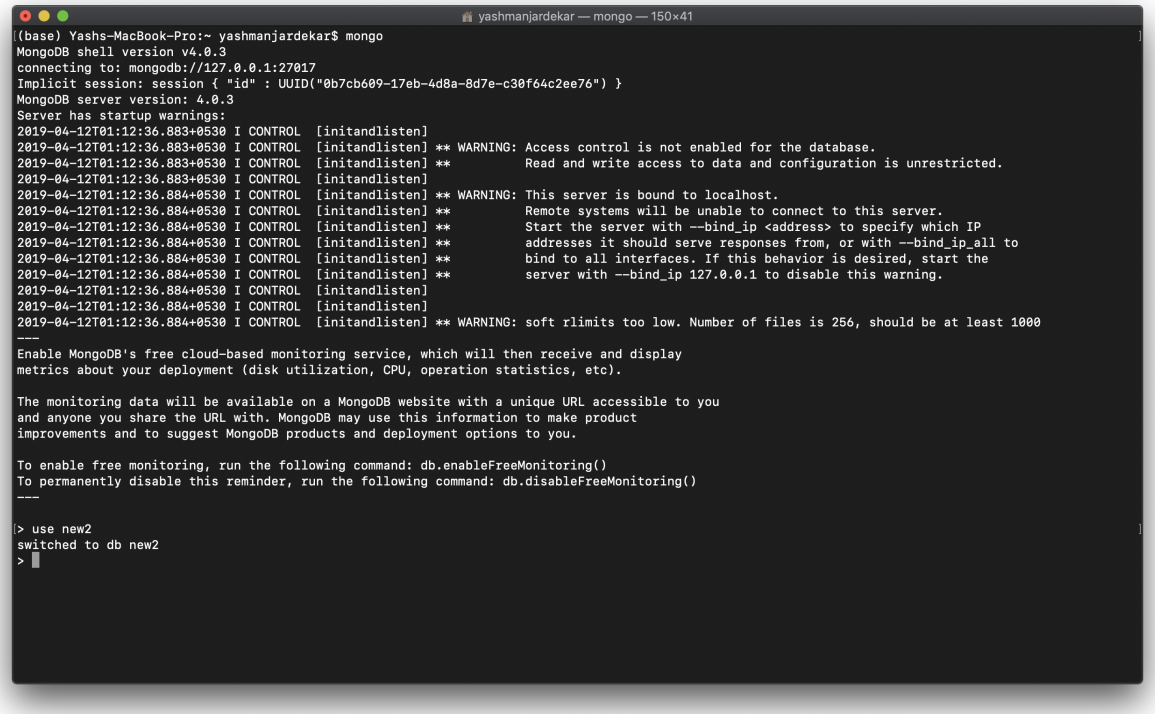

Figure 4.12: Starting mongodb and using new2 database

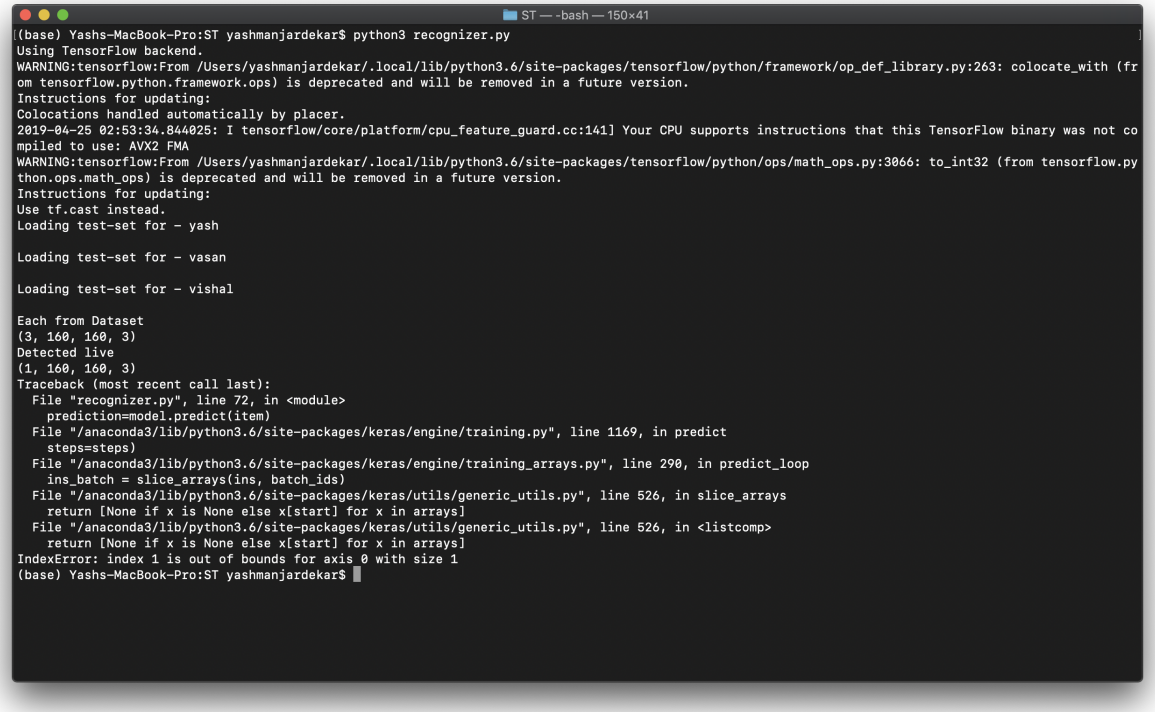

Figure 4.13: Running recogniser code to identify the person.

Chapter 5

Technology Stack

5.1 Software requirements

Softwares : Python, Keras library, Tensorflow, Mongodb, numpy, opencv, sklearn, pip3, cv2, Pymongo., Pandas.

Operating system : macOS/Ubuntu

5.2 Hardware requirements

RAM : 8GB and above

Processor : Intel Core i5 and above

Components : webbcam

Chapter 6

Unified Modeling Language Diagrams

6.1 Activity Diagram

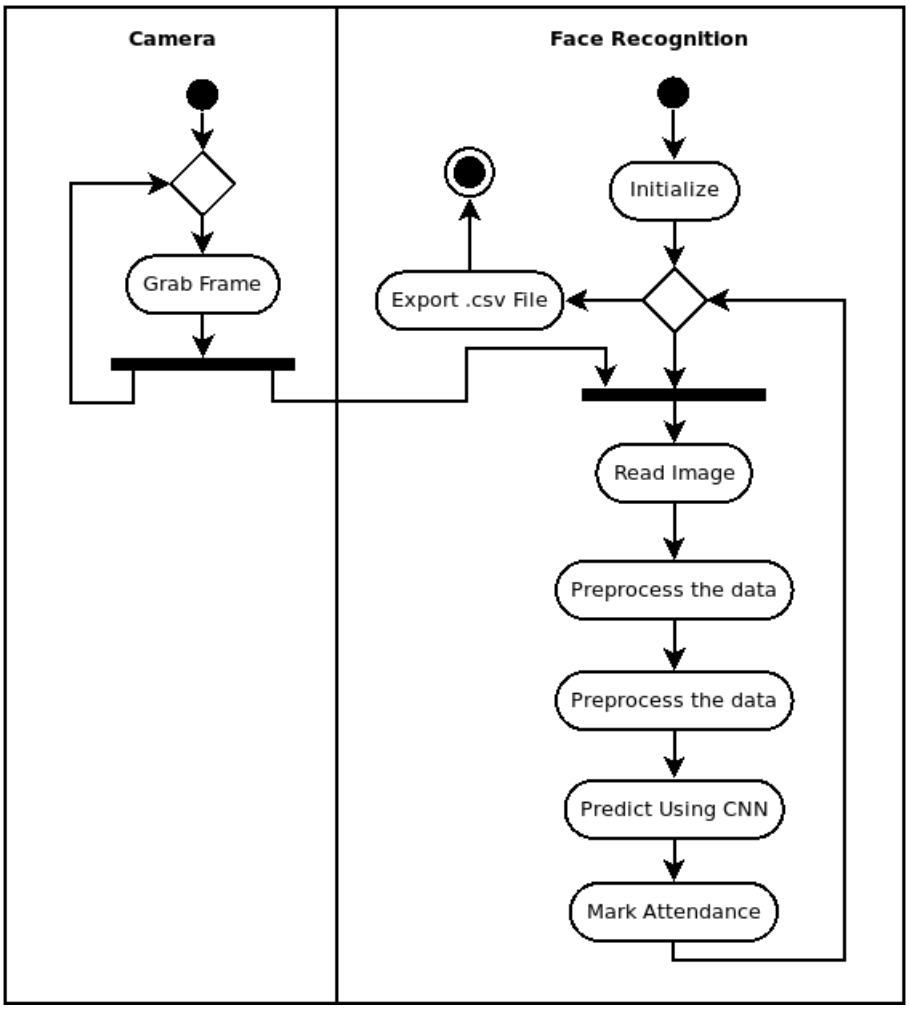

Figure 6.1: Activity Diagram

6.2 Use Case Diagram

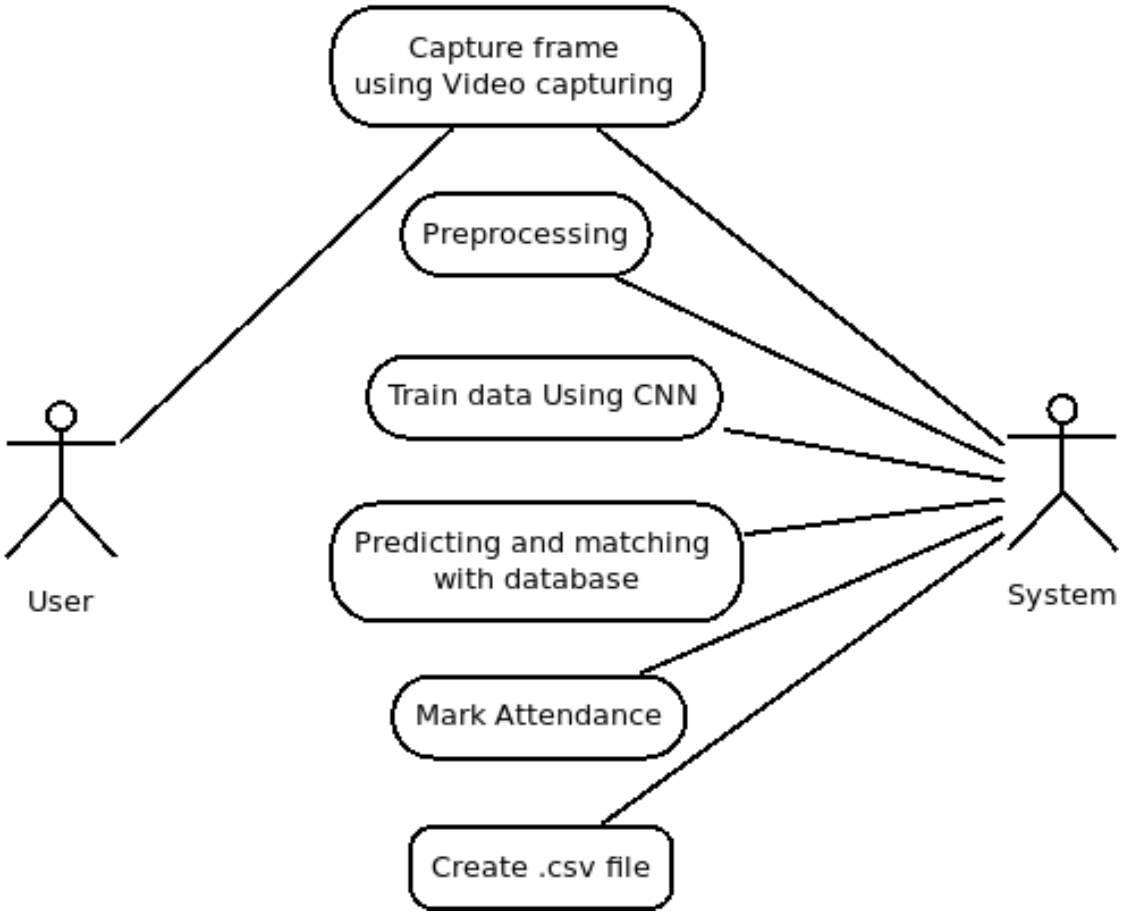

Figure 6.2: Use Case Diagram

Chapter 7

Conclusion and Future Scope

Initially the technique selected was CNN using Back Propagation Technique. The best part of Back Propagation Technique is that the accuracy of a system can be controlled up to a certain level. It is also observed that Back propagation technique takes lots of time to train a system which may decrease the usage of the system.

Compared to Convolutional Neural Network CNN using back-propagation technique, Siamese neural network are faster. Siamese network aims to outperform all available baselines by a significant margin and try to come close to the best numbers achieved by the previous networks. The network aims to outline new results comparing the performance of our networks to an existing state-of-the-art classifier developed for Our own data set. CNN are used for image classification and recognition because of its high accuracy.

Thus, after extensive research and study the team members came to conclusion to use Siamese network for Face Recognition. The Siamese network has been proved to be an efficient model for character recognition. But here while implementing the model there were some difficulties. Although we were successful in generating as well as most difficult part training the data, yet while running the "recognizer.py", we were able to get some output though it was not accurate but the system was able to recognize a single person and the .csv file was generated with the attendance in it. There was made a serious attempt so solve this problem but that to no avail.

This system has the potential to be an efficient system which can be used in various academic institutions. This will definitely reduce the hard work of both faculty and students in managing and marking their attendance respectively.

Bibliography

[1] Zuzana Bukovcikov, Dominik Sopiak, Milo Oravec, Jarmila Pavlovicov,Face Verification Using Convolutional Neural Networks with Siamese Architecture ,59th International Symposium ELMAR-2017,18-20 September 2017, Zadar, Croatia.

[2] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov,Siamese Neural Networks for One-shot Image Recognition,Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 2015.

[3] Y. H. Kim, H. Kim, S. W. Kim, H. Y. Kim, and S. J. Ko, Illumination normalisation using convolutional neural network with application to face recognition, Electronics Letters, vol. 53, no. 6, pp. 399401, 2017.

[4] Yoshua Bengio. Learning deep architectures for ai. Foundations and Trends in Machine Learning, 2(1):1127,2009.

[5] Yaniv Taigman, Ming Yang, MarcAurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to humanlevel performance in face verification. In Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, pages 17011708. IEEE, 2014

[6] Cheng, B. & D.M.Titterington 1994. Neural networks: a review from a statistical perspective. Statistical Science 9, 254.

[7] Churchland, P.S. & T.J.Sejnowski 1992. The computational brain. Cambridge, MA: MIT Press (Bradford Books).

Appendices

Detailed information about the software installations which are used in this project are showm below.

Appendix-A: Python Download and Installation

1. Go to terminal and do as following commands
sudo apt-get update
sudo apt-get install python3

2. See and set Path for Python from Terminal as
$import os
$os.environ["PYTHONPATH"]
'/home/dev/python-files'

After this save and exit.

Appendix-B: Python PIP Download and Installation

1. Go to terminal and do as following commands
sudo apt-get install python3-pip

Appendix-C: PIP Modules Download and Installation

1. Install Keras from Python3 PIP
$pip3 install keras

2. Now install tensorflow from Python3 PIP
$pip3 install tensorflow

3. Now install remaining modules from Python3 PIP
$pip3 install module-name

We will get succesfully installed prompt in our terminal. Now python and all its modules has been installed.

Acknowledgement

We have great pleasure in presenting the report on **Facial Recognition based Attendance System** We take this opportunity to express our sincere thanks towards our guide **Prof. Sachin H. Malave** Department of Computer Engineering, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Sachin H. Malave** Head of Department,Computer Engineering, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Amol R. Kalugade** BE project co-ordinator, Department of Computer Engineering, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Yash Manjardekar :
15102013 :

Sathyaveer Karmarkar :
15102031 :

Vasan Naddoni :
15102015 :

Vishal Chavan :
15102033 :