Attendance Monitoring System using Facial Recognition

Tandrima Goswami, Divyanshi Sharma, Rahul Pratyush
Dept. of CSE
MIET
Greater Noida, India

Ankit Kumar
Dept. of ME
MIET
Grater Noids, India

***Abstract*: In today's blistering pace of population, forgery is proportional to the rate and biometric is the only source to overcome it to an extent. In which there's fingerprint sensor, voice recognition, facial recognition, etc. Face recognition plays a big hand in putting a halt to all the counterfeit happening. Seeing all the circumstances, the authors came across with a facial recognition program that uses computer vision to detect an individual's face and identify it with its data and help in managing attendance system of an institution. Attendance system required human work to keep a track of every individual back in the day, with facial recognition a considerable amount of work shall be reduced as computer can manage attendance as well as keep a track of it and in an institution every individual's attendance can be managed at a single place with the help of facial recognition.**

Keywords: Facial Recognition, OpenCV, DBMS, Python, Android, JAVA

INTRODUCTION:

The industrial revolution was a period when the manufacturing of goods moved from hand production to machine production. The first three industrial revolutions which marked the advancements in our modern society were: The Steam engine, The age of science, Mass production and now it is happening for the fourth time – The Rise Of Digital Technology aka Industry 4.0 [1]

This revolution drifts towards automation and data transfer in manufacturing technologies, it. includes technological processes like Internet of things (IOT), artificial intelligence (AI), etc. This era is a fusion of physical, digital and biological technologies. Some emerging technologies includes autonomous vehicles, facial recognition systems, AR and VR software, machine learning, etc.

In 1999, at Intel Gary Bradski started with a vision with the coming age of upgraded facial recognition that will require close to null human exertion and totally computer based. As facial recognition is one of the best way of human identification, a lot of work has been done through years. Facial recognition is being used by various tech companies and for different purposes. In China people swipe their faces to make payments, nowadays its used as a biometric process to unlock your phones.[2]

As the tech advances automation can be implemented to almost all the tasks. One example is the Attendance system. In this the authors came up with an attendance system by developing an application using facial recognition, this application identifies the person's face by comparing it to the data base and increments the attendance by a variable named 'count' and uploads the attendance of the person to the server.

PLATFORM:

Machine Learning: It is a class of algorithms which is data driven and automates analytical model building. It's a subset of artificial intelligence and provides system the ability to learn from data, identify patterns and make decisions with minimal human intervention and without being explicitly programmed. The systems or models built on machine learning algorithms have the capability to learn from former experiences. Machine learning algorithm detects patterns in datasets that generate insight and helps the model to make better decision and predictions by adjusting the program actions accordingly.[3] Machine learning can be used in applications like medical diagnosis, image processing, etc.

Python: It is a general-purpose object-oriented high-level interrupted language with easy system and dynamic semantics. It was created by Guido Van Rossum in 1989 and was first released in 1991. It is an open source language which makes it free for everyone to access. It allows you to focus on the core functionality of an application[]. Python has a wide range of built in libraries which are a collection of function and methods that allows you to perform various tasks without writing your code, for example: TensorFlow, numpy, pandas, scipy, PIL, seaborn, etc. Python is used for developing websites, web apps, desktop GUI applications. AI, machine learning algorithm, mobile applications, etc.

JAVA: Java is a general purpose, strictly object-oriented high-level programming language. It was designed to have per implementation dependencies as possible. It is more complex to learn than other programming languages but Java code can be written once and executed from anywhere. Java was developed by James Gosling at Sun Microsystems in 1995. The JRE or Java Runtime Environment provides the libraries required to execute a Java application. JDK or Java Development Kit is a software development environment used for developing Java applications and applets. It includes JRE, an interpreter, java, an archiever, documentation generator and other tools required in Java development.[5] It is used for the development of web application, android application. Desktop GUI application, web servers, etc.

ANDROID:

Android Inc. was founded by Andy Rubin, Rich Miner, Nick Sears and Chris White in Palo Alto, California in October 2003. Android was started as a mobile operating system based on a modified version Linux kernel [6]. Now its being used in various technologies, not just smartphones. Android provides a rich application framework, that allows us to build advanced apps and games for mobile development in a Java language environment. Android software development is a process by which new applications are developed for the devices fulfilling the requirements of android operating system. Some android development languages are Java, Kotlin, Flutter, Python, C++, etc. Java is the official language for android app development and is most frequently used in this topic some of the most common android development environments are Android Studio, Eclipse, IntelliJ IDEA, C++ builder, etc.

PYTHON LIBRARIES:

A Python library is a reusable piece of code that you may include in your program/project. It allows you to perform various functions without writing the code. It simplifies the code and saves time. Some Python libraries used in this project are:

Numpy: Numpy is a Python library which stands for numerical python. It is the fundamental library for scientific computing in Python. It provides high performance multidimensional array and matrices object along with a large collection of high-level mathematical functions to perform on these arrays.[7] It can be used to perform mathematical and logical operation. Fourier transformations, operations related to linear algebra, etc.

OpenCV: it stands for Open Source Computer Vision. OpenCV is a library of programming functions which targets real time computer vision. It is the library used for image processing. It was originally developed by Intel and was later supported by Willow Garage then itseez. It was built to provide a common infrastructure for computer vision. It has more than 2500 enhanced algorithms.[8] These algorithms can be used to detect and recognize faces real time object detection. Extract 3D models of object, track camera, movements, etc.

Pillow: Python imaging library (PIL) or known as pillow is a library for python that provides a wide array of image processing features and is easy to use. It offers several standard procedures for manipulating images. It is used for opening, manipulating and saving different image file formats.[9] Some operations which can be performed using this library are cropping, resizing, adding, text to image, greyscalling etc.

Pickle: Python pickle module is used for serializing and de-serializing a python object i.e., it converts a python object into a byte stream to star it in database or transfer data over the network.[10] This process is called serialization and the revers process is called de-socialization.

TensorFlow: It is a free open source library. It is a symbolic math library used for machine learning application such as neural networks and deep learning. It manipulates data by creating a DataFlow or a computational graph[11]. It consists of edges and nodes which are used to perform operations and do manipulations. TensorFlow is now widely used to build complex deep learning models.

DATABASE:

It is a systematic collection of data and info that can be easily manipulated. In relational database, digital information about a specific user is organized into rows, columns, tables which are indexed to make it easier to find relevant info. through SQL queries. The ability to control read/write access or analyze usage is done by the database manager.

SQLite is a C-language library that implements a small, fast, self-contained, high reliatrility, full featured SQL database engine. It is built int all mobile phones and most computers. The SQLite file format is stable, cross-platform and backwards compatible. It is used as containers to transfer rich content between systems. Its source code is in public domain and is free for everyone.[12]

Database-with android: android.database.sqlite is the main package that contains the classes to manage your own database. SQLite open helper class provides the functionalities to use the SQLite database. Mainly there are two constructors of SQLite helper class, SQLite Open Helper (context context, string name, SQLite Database, CursorFactory factory, int version), SQLite Open Helper(… int version, Database Error Handler error handler)//Specifies error handler[13]

Database development: The pictures of each and every individual are arranged into separate folders according to their details. When a person’s image is captured, the recognition algorithm is applied on the person’s face. The image is cropped according to the given height and width and then is converted into greyscale image is watched with the pixels of the images stored in the database. Whenever there’s a watch, the count variables are incremented by one and the person’s attendance is worked. In our android application there are separate databases for student, teachers and courses which is managed by the admin.

FACE DETECTION AND RECOGNITION:

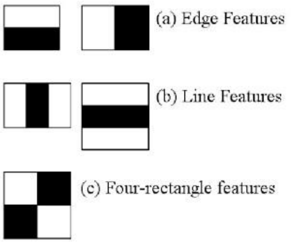
Facial Recognition and Facial Detection are often used as similar terms but works in completely different areas. The main purpose of the program is to identify a person’s face by identifying it and matching it with the images captured at the time of recognition used the pixels of the image to match it with its data. Variance in pixels will result in a different outcome.

Face Detection: Some of the common features of human faces are specific location of eyes, nose, mouth, bright nose bridge region, etc. These are called Haar Features.

This method is proposed by Paul Viola and Michael Jones in their paper, “Rapid object Detection using a boosted cascade of simple features” in 2001. For training the classifier we need a lot of positive and negative images i.e., with and w/o faces. For extracting the features, where each feature is single valued is done by subtracting the sum of pixels under white rectangle from the sum of under black rectangle. For calculating each feature we need to find the sum of pixel under white and black rectangle.[14]

Most of the features are irrelevant. Selection of the best feature is done by “Adaboost”.

When each and every feature is applied on the training image, a threshold is generated which classifies the face into positive and negative. The features with minimum error rate is selected. At the beginning each image has equal weight after each classification weights of misclassified images are increased. This process is continues until a regional accuracy or error rate is achieved.



If an image contains non face region it is better idea to have a simple method to check if a window is not a face region. If it is not then discard it right away. For this there’s a concept of cascade of classifiers. The features are grouped into different stages and is applied one-by-one. If the window fails the first feature we discard it else we will move to the second stage.

OpenCV comes with a trainer as well as detector. We can train our own classifier for any object. It contains pre-trained classifiers for face, eyes, etc. Those XML files are stored in opencv/data/Haarcascades/folder[15]

We need to load the XML classifier

```
Import numpy as np

Import cv2

Face_cascade=cv2.CascadeClassifier('cascades/data/haarcascade_frontalface_alt2.xml')

Eye_cascade=cv2.CascadeClassifier('cascades/data/haarcascade_eye.xml')
```

If faces are found, it returns positions of detected faces as rect(x,y,w,h).

Once we get these locations, we create a ROI (Region of Interest) for the face and apply eye detection on this ROI

```
faces=face_cascade.detectMultiScale(gray,scaleFactor=1.5, minNeighbors=5)

   for(x, y, w, h) in faces:

       #print(x,y,w,h)

       roi_gray = gray[y:y+h, x:x+w]

       roi_color = frame[y:y + h, x:x + w]

      eyes=eye_cascade.detectMultiScale(roi_gray)

       for(ex, ey, ew, eh) in eyes:

          cv2.rectangle(roi_color,(ex, ey),(ex+ew, ey+eh), (0, 255, 0), 2)

cv2.imshow('frame', frame)

cv2.destroyAllWindows()
```

Face Recognition: It is the process of identifying or distinguish the identity of a person using their face. It captures, analyze and compares patterns based on the person’s facial details.[16] Facial recognition can be performed very easily due to the very famous python library created by Intel known as OpenCV. It has various algorithms which makes it easier to work on Image processing data.

Facial detection works in 90%-95% of clear photos of a person where as facial recognition is only 30-70% accurate. This makes Facial recognition less reliable on accurate thus facial detection. Using OpenCV we can identify a person by comparing the image captured by a web cam (or any other camera) to the images of that person present in the database OpenCV requires a huge database. The accuracy of the model increases with the number of pictures present in the database to compare it with the original image. There are other factors that influence the accuracy of the model. The model is ‘lumination dependent’[17] which mean that it is sensitive towards lightning and would probably not recognize the person in a dark room with the precision compared to the picture of a person in a bright room. It also locks the ability to identify a person’s face from different angles, increased contrast of shadows, blurry image or person wearing glasses with perfection.

To fix most of these problems it is important to use a good image preprocessing filter before using facial recognition. You should remove pixels around the face which are not being used and sticky solely to the region of interest.

Image and its matrix: We convert the colored images into grey scale.

```
gray =cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

The computer reads all the images into grayscale which is present to it by pixel matrix. Each pixel is represented by a matrix element ranging b/w the integer from set {a,1,2,……,255}. The elements represent the pixel values changing from zero (black pixels) to 255 (white pixel)

This colored images are represented with three grayscale image matrix (for each color component-red, green, blue (RGB) [18]

After applying the Harr cascade classifier and detecting the face, the model compares the captured image with the database. It does so by comparing the pixels matrix of the captured image by the pixel matrix of the image present in the database. It identifies the person based on the highest match in their pixel matrices.

```
faces=face_cascade.detectMultiScale(pics_array,scaleFactor=1.5, minNeighbors=5)

    for (x, y, w, h) in faces:

                roi = pics_array[y:y+h, x:x+w]

                x_train.append(roi)

                y_labels.append(id_)
```

It is very important to remove the unnecessary pixels from the image such as hair color, background, inner face region, etc. as they prevent the model to make predictions accurately. It is also important to resize the image to the standard size or the model with identify the person with the person's image present in the database with the same size.

LBPH Algorithm: The local binary pattern histogram (LBPH) algorithm is a simple solution on facial recognition problem which works better in different environments and light conditions than most of the other facial recognition algorithms. It can also recognize both front face and side face. To use this algorithm we need to create an intermediate image that describes the original image by highlighting the facial characteristics [no]. the dataset can be created by taking many image samples of a singe person and give a unique ID or name to the person in the database. In LBPH labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. The image will be represented as a 3*3 matrix containing the intensity of each pixel (0~255) we take the central value of the matrix as threshold which is used to define new values from the 8 neighbors[no]. This process will give a new image which represents better characteristics of the original image. [19]

```
recognizer = cv2.face.LBPHFaceRecognizer_create()
```

After identifying the identity of the person using their faces it displays the name of the person on the top of the rectangular box which shows the region of interest (ROI) of the person's face. If the face is not recognized by the model it displays the message as 'face not found'.

ATTENDANCE SYSTEM:

There are various examples for using the facial recognition model. One of the example is the attendance system. In this project the author has used facial recognition to automate the Attendance process in schools, colleges or firms. Its is an android mobile application which automatically marks the attendance of a person by identifying them using their faces and uploads it to the server. There is a database which contains the details of the students, teachers, or employees and it is controlled by the administrator. The admin ID can access all the data of the students or employees and it can also be used to enter a new dataset in the database through the application. The students or employees are also provided with their own ID's incase they want to check their attendance. The application is made by using a well known programming language called Java and the environment used is Android Studio.

Android Studio has various built in tools which make it easier for the developer to create application one can easily drag and drop various methods without actually writing the code for it. it is very easy to use and learn. Android studio provides a built in helper library known as SQLite which is used to insert a database into the project. It has various classes on functions which makes the process easier for the developer.

RESULT:

We have successfully created an android mobile application for attendance monitoring system using Facial Recognition. Working on this project given you a proper knowledge of Machine Learning algorithms, python libraries, Java and Android. We have achieved our goal of developing an facial recognition system using OpenCV which gives us around 30-70% accuracy depending upon the no of pictures present in the database which is used by the model to compare it with the real time faces and identifying them. The LBPH algorithm used does not provide the accuracy which could be provided by other algorithms such as Eigenfaces, PCAC principle component algorithm, fisherface algorithm etc.[20] but the LBPH algorithm provide firm results depending on training and testing datasets and it is very easy to learn and use and it works better in different environments and light condition.

CONCLUSION AND FRAME WORK:

Facial recognition has been a aspiring topic for many developed for a long time. To achieve good accuracy in a facial recognition model one must keep on experimenting with the data and algorithms. To improve a facial recognition model one can input more number of pictures of a person into the dataset. It allows the model to compare with a wide variety of pixel matrices. One can also input pictures having different environment settings with different lighting it will help the model train in different environments and increase precision. One can also enter images of a person from different angles so that the model can identify a person with both its side face and front face.

To improve the model more complex algorithms must be used like PCA, Eigenfaces, 3D face recognition, SVM, etc.

The most popular method of facial recognition is neural networks and deep learning. Neural networks are used to recognize the face through learning correct classification of coefficient calculated by the eigenface algorithm [21]. facial recognition is achieved using deep learning’s sub field that is Convolutional Neural Network (CNN) which is a multilayered network trained to perform a specific task using classification.

Further, there is a lot of development possible in this area which will narrow the gap between the facial recognition system and will increase accuracy and efficiency of the model.

REFERENCES:

[1] Wikipedia industry 4.0
[2] facial recognition revolution-boyden.com
[3] mathworks.com
[4] medium.com
[5] Java, technopedia.com
[6] Android Wikipedia
[7] numpy tutorintpoint.com
[8] quora algorithms in OpenCV
[9] Image processing in python with pillow, autho.com
[10] understanding python pickle, geeksforgeeks.com
[11] tensorflow library quora.com
[12] sqlite.org
[13] searchsqlserver.com
[14] OpenCV python tutorials
[15] towardsdatascience.com
[16] facial recognition, gemalto.com
[17] facial recognition using OpenCV NVIDIA (research paper: Shervin EMAMI1, Valentin Petruț SUCIU2)
[18] image and its matrix, matrix and its image (research paper: Vesna Vučković)
[19] facial recognition: understanding LBPH algorithm, towardsdatascience.com
[20] facial recognition for beginner, towardsdatascience.com
[21] facial recognition using neural networks ieeexplore.ieee.org