Face Recognition based Attendance System

Team Number :

Roll Number	Name
2019201063	Mudit Malpani
20161072	Sourav Kumar
20171192	Prazwal Chabra
20171154	Manoj Attanti
2019201059	Vikram Keswani
2019201062	Nabhiraj Jain

Overview

This software product is a tool for automating the attendance system in college.
Manual Attendance has been the norm for a very long time. Teachers invest at least 10-20%
time of class time in taking attendance and this is done for every class, every day.
So here, we present a small contribution to automate the college attendance system using state
of the art **Computer Vision** and **Deep Learning** techniques.
In order to make our product highly available and scalable, data will be processed at multiple
nodes instead of putting the load on a single processing unit - **Distributed System**.

This document covers the High-level design of the project.

In the very first step, the source will send the class image to our system. The source can be a camera installed in a classroom or image/video clicked by a designated person. Then, after this first task on our table will be to extract the individual face images from a class/group image. These images are then passed to the processing nodes via the load balancer unit.

In processing units, captured faces will be matched with the individual face id of the student in the database. If the faces match, the student is marked present in a separate database. For this task, we will make use of **Convolutional Neural Nets(CNN's).** CNN's are great for capturing important face details or features and thus can be utilized for distinguishing among students.

Modules Involved-
Input Unit - Gets the group picture/video and extracts individual images. Another job is to fetch the images of all students from the database of the corresponding class.
Load Balancer Unit - Distributes the workload to different processing units.
Data Processing Unit - To process the images and find the corresponding student.
Output unit - Assembles the output from the processing unit and finally marks the attendance in the database.

Terminology

- **Face extraction**: the process of detecting all the human faces present in a image and extracting them out in a specific format.
- **Face Recognition**: process of recognising the person to whom this face belongs.
- **Class :** group of people who are registered for a course
- **Group Image** : image of the class (present on the given day) taken as a whole.
- **Item image** : image of individual student extracted from the group image.
- **Class Info** : all the information about that class stored in our database.
- **Class image set** : collection of images of students present in that class (stored in our database, this will be used for matching the images)
- **Class Item image** : the image of an individual student taken from the class image.

Note :: item image and class item image are indexed like item_image[i] can be mapped to class_ item_image[j] where i and j may or may not be the same.
For example if 20 student are present in one day there will be following number of item image
Item_image[0], item_image[1] ……………… ,item_image[19]
And in a class (say SWE) there are 100 students then the class item image will be.

Class_item_image[0],Class_item_image[1] …………………….,Class_item_image[99]

Group Image (input to the system) will be divided into a number of item images and will be compared with the images present in class images set to mark the attendance of the students who are present.
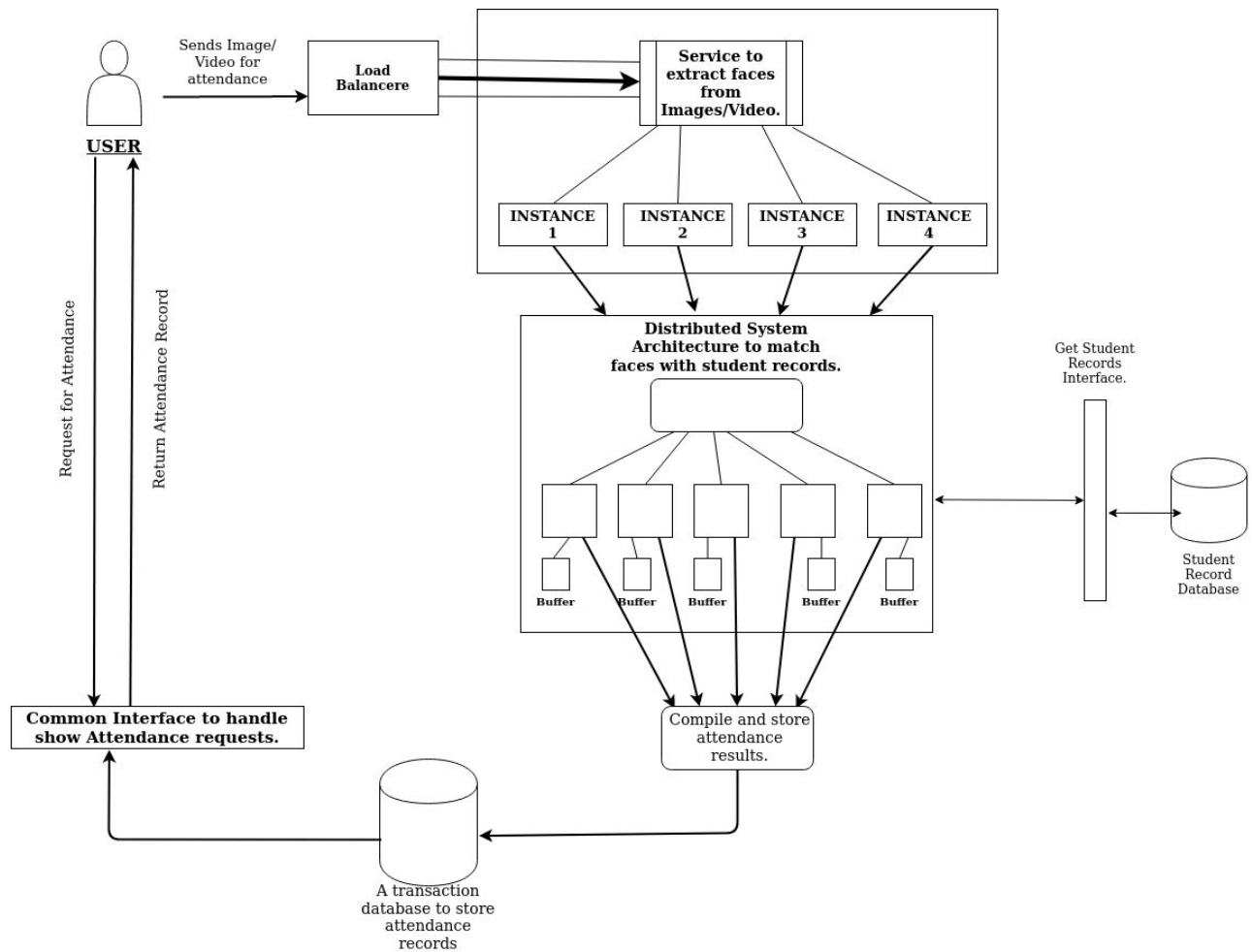

Tenets:

The main working principle of the project is to capture the group image of students in the hall, detect and recognize them as individuals. Further the recognized image of the student is provided with attendance, else the system marks the database as absent. It also include to help the lecturers to improve and organize the process of track and manage student attendance and absenteeism

The main objective of introducing the Student attendance facial recognition system is to
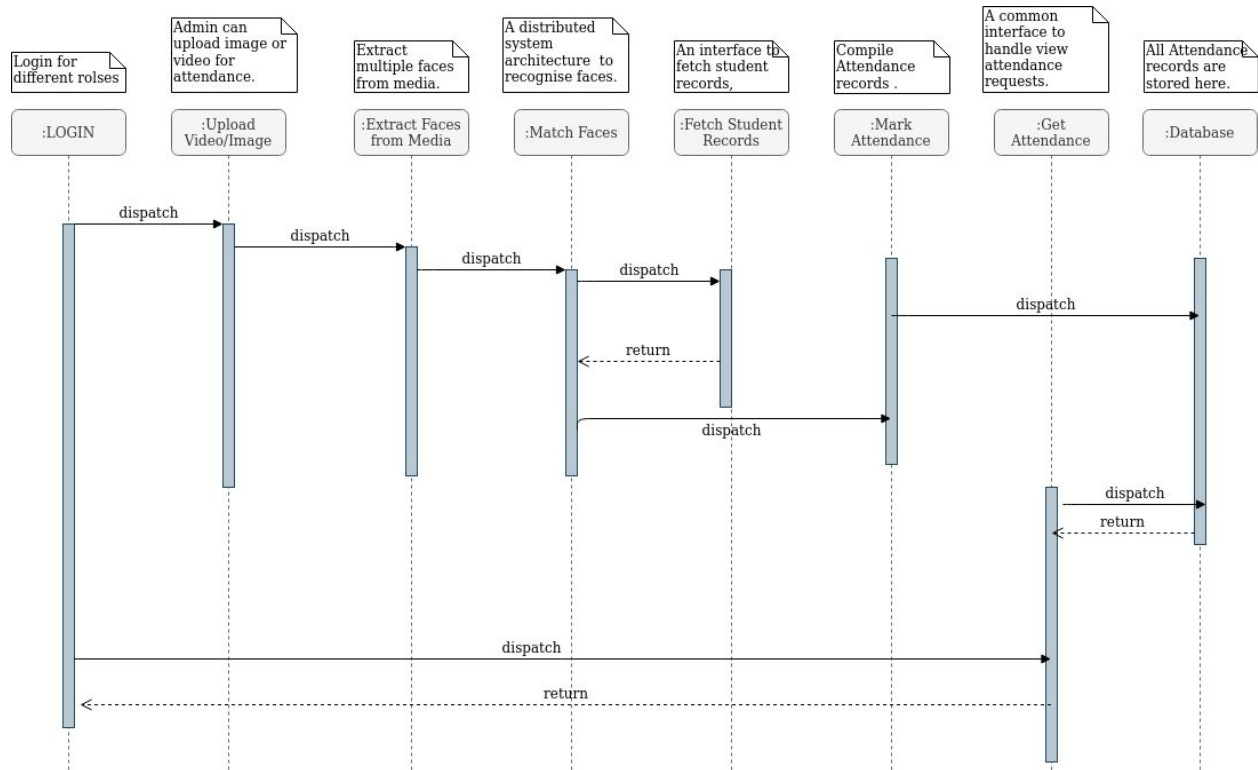- Convenient : Reduce manual process errors by providing automated and a reliable attendance system uses face recognition technology.
- Used for secure purposes: Increase privacy and security which a student cannot present himself or his friend while they are not.
- Calculate absenteeism percentage and send reminder messages to students.
- Automated update in the database without human intervention.
- Produce daily reports for the lecturer and send to the individual students and teacher for respective subjects.

Attendance management system is a necessary tool for taking attendance in any environment where attendance is critical. Most of the existing approaches are time consuming,intrusive and required manual work from the users. In this project, we eliminated the above challenges by demonstrating the use of face recognition in students.

Proposed Architecture

USER

Sends Image/Video for attendance

Load Balancere

Service to extract faces from Images/Video.

INSTANCE 1   INSTANCE 2   INSTANCE 3   INSTANCE 4

Distributed System Architecture to match faces with student records.

Get Student Records Interface.

Student Record Database

Buffer   Buffer   Buffer   Buffer   Buffer

Request for Attendance

Return Attendance Record

Common Interface to handle show Attendance requests.

Compile and store attendance results.

A transaction database to store attendance records

Sequence Diagram

Detailed Design of Components

In the above proposed architecture, there are components that will increase the availability, scalability and efficiency of our product. Following is the detailed functionality of these components:

- **Load Balancer:** In real life scenario, there can be a case when multiple classes are running simultaneously so giving the complete job to a single processing unit will be time consuming. Thus, to make our product more efficient and responsive this component is introduced which divides the heavy task into multiple processing units. Major task are:
  - Extracting faces from a single image.
  - Find the most similar face from our database.

Note:: This will also take care of reassigning jobs to other processing units if one of the processing units fails or fails to do its jobs in a given time.

- **Processing Nodes:** Once our Face recognition model is done with its job i.e extracting face from a single class image. Then these extracted images will be passed to the load balancer and it will assign a task to the processing unit by sending some face images. So, this component will be using some CNN architecture to find the most similar image from the students image database using facial details or features to the given image.

- **Read Optimised Database:** This database can only be accessed by processing units for the security of our product and have all the images of students stored locally that are mapped to corresponding student roll number. Also, will have the information regarding the students that are enrolled for a particular course to decrease the search space i.e. number of images to find most similar for a given image.

- **Class Information Buffer:** Processing nodes will be using the database interface to match faces but using this interface will make many network operations which will be costly therefore each processing node will maintain a buffer of class image information so that network traffic to the database can be minimized. So the request will go to the database only if it's a miss in-class info buffer. This will increase the overall speed of the application by reducing the amount of network operation.