CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Being a literate human being we all know how attendance plays an important role in a student life. The main purpose of attendance is keeping record of the students in an organization. Generally, it refers to the act of being present at that moment in an organization.

Attendance is arguably one of the major important indicators of a student in school life. Attendance is also powerful predictor of student's performance and outcome of the results. Not only in the school's but also in the workplace (or) social events, attendance plays an important role in determining the individual's discipline towards their respective organizations.

Consistent attendance of the student shows us the commitment and responsibility, which are basic essentials for the student's career growth. Now the next thing is we know how attendance is being taken in the schools. the general and traditional way of considering the attendance is by calling the roll no's and the names of the students by teachers.

As increased innovation in the technology nowadays there are various methods (or) ways are there to consider the attendance. For example, biometric identifiers which identifies the finger print, face recognition, palm print, geometry of the face, iris recognition etc. it is the fact that the general way of considering the attendance looks very cheaper and easy but here comes the problems like wastage of time, basic calculation errors in attendance percentage, roll call of the students can be missed etc.

To overcome all these problems which occurred in general method of considering the attendance we go the advance method of considering the students or employees attendance. In this paper we discuss about the attendance management system based on face recognition. attendance for the students is considered through mobile application in which the students should submit their respective attendance by scanning their faces using their individual mobile camera.

Firstly, Face recognition which refers to the process of giving machines, tools, and software the ability to identify the different facial features of an individual. and The Machine learning algorithms quickly capture faces, collect and retrieve different facial features which also matches them with pre-existing pictures to form a connection.

The extraction of face could include the outside border of the eyes, top of the nose tip, bottom of the chin. Then, the Machine Learning algorithm is repeatedly trained using different data points to locate these points on the face and turn them towards the center to align to match the database.

Majorly there are different types of face recognition algorithms:

- Eigenfaces
- Local Binary Patterns Histograms (LBPH)
- Fisher faces
- Scale Invariant Feature Transform (SIFT)
- Speed Up Robust Features (SURF)
- Convolution neural network (CNN)

A Convolutional Neural Network is a type of Deep Learning neural network architecture which is commonly used in Computer Vision. it is a field of Artificial Intelligence that enables a computer to understand and interpret the visual data. When it comes to Machine Learning ANR (Artificial neural network) perform really well. CNN is used in various datasets like images, text, speech etc…. Various types of Convolution Neural Networks are used for various purposes, for example for predicting the sequence of words we use RNN (recurrent neural network) more precisely an LSTM, and also for image classification we use the (CNN) Convolution Neural networks. In a simple terms Convolutional Neural Network (CNN) is a special type of neural network which is designed to process the data through multiple layers of arrays. It is well suited for the applications like face recognition

CNN Architecture

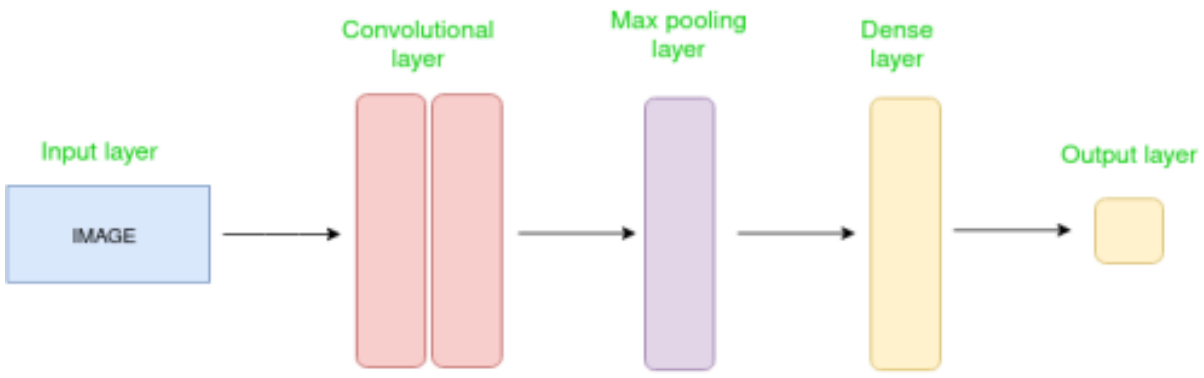


Figure 1.1 CNN Architecture.

Convolutional Neural Network consists of different layers where the first layer is the input layer then followed by Convolutional layer, Pooling layer & fully connected layers (output layers).

Generally, the Convolution neural network (CNN) uses a mathematical operation called the convolution. Here we define convolution as a mathematical operation on which the two functions f and g will produces the third function. The third function will express how the shape of one graph is modified by the other.

In Convolutional Neural Network convolutional layers are the main fundamental blocks that make the desired output happen. In a particular image recognition application, a convolutional layer is made up of several filters (Kernels) to detect the various features of the image. Now we see how this work is best illustrated with an example.

When we see someone walking towards us from a certain distance. Our eyes will try to detect the edges of a particular person & we then try to differentiate that particular person from other objects like buildings, vehicles etc. As the person comes towards us, we mainly try to focus on the outline of the person and then we try to deduce if the person is male or female, Thin or fat etc. And finally, if the person gets nearer our focus shifts toward other features of that particular person, such as his facial features like whether if he/she is wearing specs, Mask etc. Concluding that generally, our focus shifts from broad features to particular features.

Likewise, in a Convolution neural network (CNN), we have Various layers containing various kernels(filters) in charge of detecting a specific feature of the target that we're trying to detect. The very first layer tries to focus on broad features as we mentioned in the above example, then the very next layers try to detect very particular/specific features. In a CNN, the values for the various filters in each convolutional layer is obtained by training on a specific training set. In the end of the training, we have a unique set of Kernel values which are used for detecting the specific features in the dataset. Using this set of Kernel values, we then apply them on new images so that you can make a prediction about what is contained within the particular image.

The below image shows a typical CNN network. The very first few convolutional layers (conv1 to conv4) detect the various features in an image such as lines, edges, shape etc. The final few layers are used to classify the result Whether the image contains faces belong to person 1,2 or 3.

A Pooling layer in CNN is used to reduce the size of the image and to boost up the calculations, and also to make some of the features it detects a bit faster and more efficient.

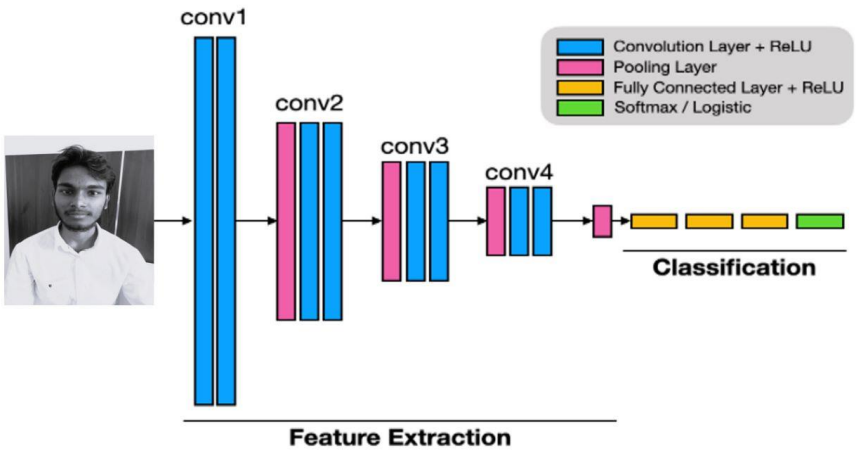


Figure 1.2 Convo Layers.

Haar Cascades

Haar cascade is an algorithm that can detect objects in images, irrespective of their scale in image. This algorithm is not very complex and can run in real-time. We can train a haar-cascade detector to detect various objects like buildings, cars, persons, chairs etc. This Haar Cascades uses the Cascading windows it tries to compute features in every window and finally it classifies whether it could be an object or not.

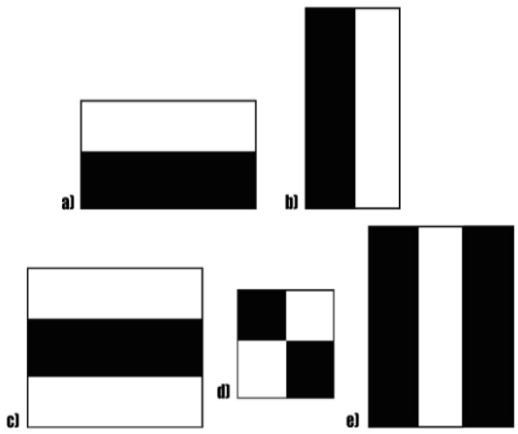


Figure 1.3 Types of Haar features.

The Haar algorithm can be explained in four stages:

- Calculating Haar Features

- ➢ ➢ Creating Integral Images
- ➢ ➢ Using Adaboost
- ➢ ➢ Implementing Cascading Classifiers

A. Calculating Haar Features

The Initial step is to collect the Haar features. A Haar feature is essentially calculation which is performed in the adjacent rectangular regions at a particular location in a detection window. This calculation involves addition of the pixel intensities in each region and calculating the differences between the sums. Here are some examples of Haar features below.

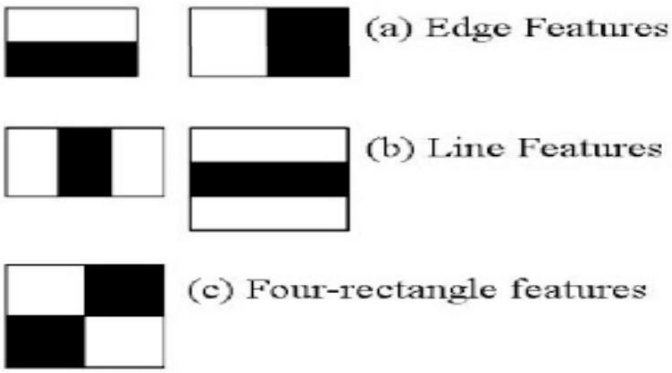


Figure 1.4 Types of Haar features.

These features can be difficult to determine for a large image. This drawback is where the integral images come into picture because the number of operations is reduced using the next step i.e. (integral image).

B. Creating Integral Images

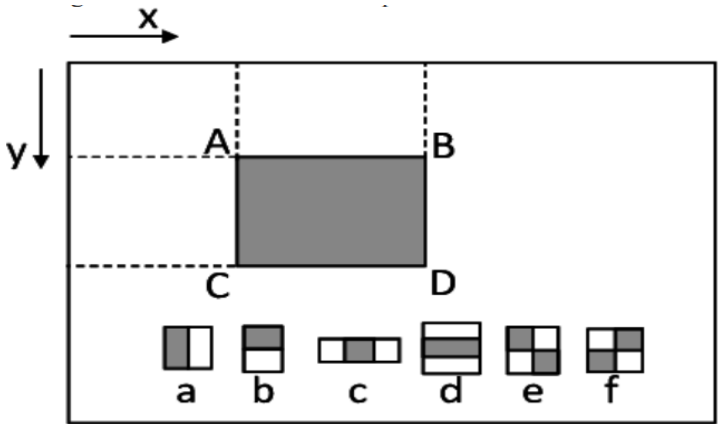


Figure 1.5 Internal Images.

In this Integral Images there is a mathematics without going into too much depth of mathematics behind it the integral images are essentially boost up the calculation of these Haar features. Instead of computing at each and every pixel, it instead creates sub-rectangles in it and also creates array's of references for each of those subrectangles and then are used to compute the Haar features.

Here an important thing to note that nearly all of the Haar features will be irrelevant while doing object detection, because the only features that are important are those of the object. Then how do we determine the best features that represent an object from the hundreds of thousands of Haar features. This remains as a question and also as a drawback That's where Adaboost comes into the picture

C. Adaboost Training

This Adaboost essentially chooses the best features and trains the classifiers to use them in an effective way. This is a combination of two classifiers they are "weak classifiers" to create a "strong classifier" that the algorithm can use to detect objects. Weak classifiers are created by moving a window over the given input image, and computing Haar features for each subsection of the image. Here, the difference is compared to a learned threshold that separates non-objects from the given objects. Because these are "weak classifiers" a large number of Haar features is needed for accuracy to form a strong classifier for a reliable output.

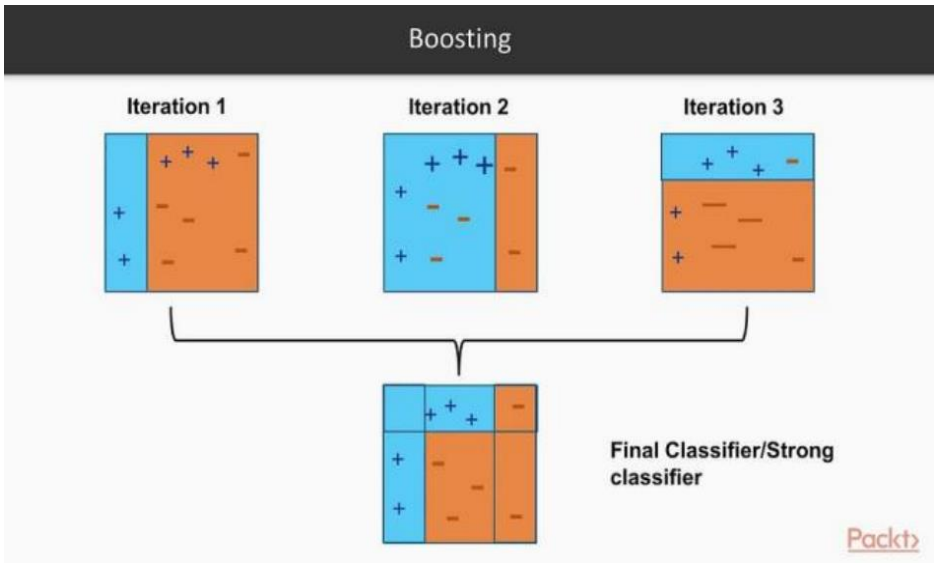


Figure 1.6 Adaboost Training.

The last step combines these weak learners into a strong learner using cascading classifiers.

D. **Implementing Cascading Classifiers**

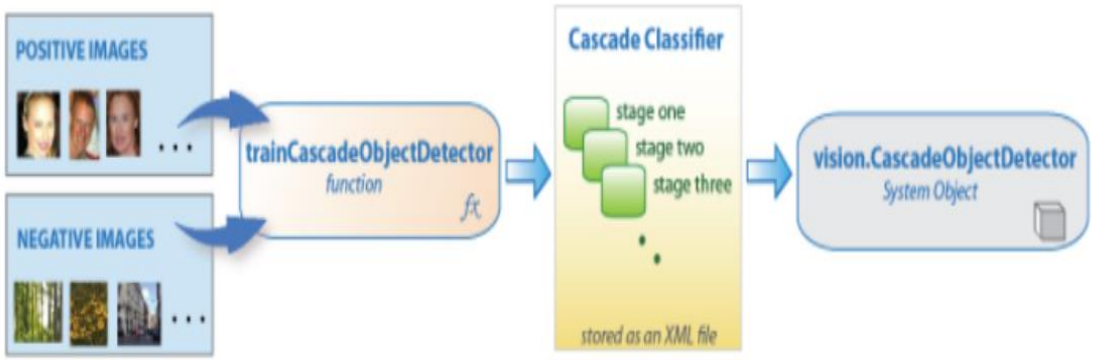


Figure 1.7 Implementing Cascading Classifiers.

The final cascade classifier is made up of a series of stages, in which each stage is a collection of the weak learners. These learners are trained using boosting or speed, which allows for a highly accurate classifier from the basic prediction of all weak learners. Based on this assumption, this classifier either decides to indicate an object was found positive or move on to the next region negative. These Stages which are generally designed to reject negative samples as fast as possible, because a majority of the windows do not contain anything of interest. It is important to maximize a low false negative rate, because classifying an object into or as a non-object will severely impair your object detection algorithm. One major thing to note about the Haar cascades is that it is very important to reduce the false negative rate, so we should make sure to tune hyperparameters accordingly when we train the model.

CHAPTER 2

LITERATURE STUDY

M. S. Mubarak Alburaiki, G. Md Johar, R. A. Abbas Helmi and M. Hazim Alkawaz. Every educational institution has its own way of marking the attendance to the students. Most often we see roll callbased attendance in our daily life. In this project the attendance for the students is considered through mobile application in which the students should submit the attendance by scanning their faces using their individual mobile camera, along with their current location. Firstly, Face recognition which refers to the process of giving machines, tools, and software the ability to identify the different facial features of an individual. and The Machine learning algorithms quickly capture faces, collect, and retrieve different facial features which also matches them with pre-existing pictures to form a connection. In this proposed project we are using there are three main components. Firstly, automatic face detection and analysis using mobile cameras of an individual. Secondly, we use face recognition API (Application program interface) which uses the machine learning algorithm and the final component is which maps the API. The extraction of face could include the outside border of the eyes, top of the nose tip, bottom of the chin. Then, the ML algorithm is repeatedly trained using different data points to locate these points on the face and turn them towards the centre to align to match the database Using this proposed system, we are having several benefits Such as it reduces the possibility of fake attendance because the system verifies the individual's identity through face recognition and then it eliminates the need for paper-based or rollcall attendance. which can be time-consuming. Finally, it provides real-time data on attendance, which can help organizations and educational institutions to manage attendance more effectively. In this project they used the (HOG) Histogram of Oriented Gradients. One of the most popular, efficient and successful "person detectors" is the Histogram of Oriented Gradients with LBPH (a type of machine learning algorithm for the classification). After conducting the testing process in educational institution, the project result shows face recognition using ML algorithms has achieved a high accuracy of detecting students faces even in a dull environment condition. It has been revealed that over 80-85% of the students are satisfied with the face recognition process. The use of the machine learning algorithms makes the proposed system more efficient, accurate, safe and secure[1].

Q. Y. Tan, P. S. Joseph Ng and K. Y. Phan. As we all know the Attendance is an essential aspect of academic and also for the professional life, and the importance of attendance cannot be overstated. Attendance refers to the act of being present of a student at a designated location. JomRFID Attendance Management System is a system which uses the RFID (Radio Frequency Identification) technology to manage and consider the attendance in organizations and educational institutions. Nowadays The RFID system is very common in various industries and also the RFID system is deployed within various fields like transportation, medical and many more fields. RFID is widely emerging in the world due to its attractive features such as good reading ranges, high data efficiency, high reliability and its low cost. It is a is a modern way of attendance management system that uses RFID technology to manage attendance in organizations and educational institutions. RFID reader contains an antenna that emits the radio waves in which the tag responds by sending back data stored in it to the reader. JomRFID is very user-friendly and very convenient to every individual. students can easily scan their ID cards to record their attendance in school or college. Hence, by the conclusion we can say that the respective faculty can save up the time on the attendance when taking it manually. As a result, by implementing the RFID system for considering the attendance in institutions makes it very simple and efficient [2].

M. R, M. D and R .P. In this paper we can see that attendance is considered using biometric in the classrooms. Managing the attendance of each and every student during the class time has become a very difficult challenge. The ability to calculate the attendance percentage of student becomes a major task because manual calculation produces many errors, also wastes a lot of time and sometimes the papers might be missed. Here the biometric involves using unique physical or behavioural characteristics such as fingerprints, facial recognition, or iris scan to identify individual student or employee. This Classroom Attendance Monitoring Using Biometric has also been carried out by the software that uses passwords for authentication purpose. The main objective of the project is to save the time. After conducting the survey, we got to know that manual attendance system takes a time for eighty students was approximately 18 seconds while when the attendance was taken by using biometrics it just took 4 seconds of the time, we can see the major difference here. After considering the attendance by using biometric the system successfully took the attendance during classes and examinations. This system successfully captured new fingerprints of the students which has to be stored in the database and the scanned fingerprints placed on the device sensor and then it compared against those previous fingerprints stored in the database successfully [3].

R. P. Vandana, P. S. Venugopala and B. Ashwini. As we know that the technology has been emerging day by day. In the emerging world not only the technology but also the education system has been reached to the new destination due to the introduction of the concept called smart classroom. As all the classrooms moved to dustless and very user-friendly in the new era of smart classroom. In this paper they discussed about the attendance is considered without the human interference. Neural network-based biometric attendance system which provide an efficient and very accurate way of monitoring the student's attendance. neural network is one of the types of machine learning algorithm which learns the patterns from the data and makes the predictions based on the learning. One of the major rules for the neural network based biometric attendance system is that it should relevant to the laws and regulations related to data privacy and security. Proper safeguards must be put in place to protect the student's personal information and the biometric data and also it should ensure that it is not used for any unauthorized or unethical purposes. In this proposed system a camera is fixed within the classroom then the camera will capture the image. the faces are detected and they are compared with the faces which are stored in the database and finally the attendance is marked. This system also proposes image-based face liveness detection method for discriminating 2-Dimension paper masks from the live faces of the individual person. Freely available or open-source resources machine learning and deep learning tools are available like dlib, Keras are most popular for making the face recognition faster and reliable one [4].

P. Sarath Krishnan and A. Manikuttan. In the recent years Facial Recognition have a remarkable upgradation and it became one of the popular features of security. we usually see it in mobile phone to unlock the device. Attendance Management System using Facial Recognition is a kind of software application which uses an advanced facial recognition technology to accurately take the attendance of a student. Generally, LBPH is a most commonly used feature for an extraction technique in computer vision, especially in the facial recognition field. It is an extension of the Local Binary Patterns algorithm and also uses a histogram of Local Binary Patterns codes to represent the image. This system has been proposed because the first and foremost thing that is happening in the schools and colleges is responding for the roll calls twice or missing the roll call of a student. Hence to eradicate this problem as well as it also maintains the records which is very tough task. After Considering the face recognition of an individual student the data is aggregated into the database. In this paper an automated real-time attendance management system using face recognition technique is used to reduce the human interference or dependency and thereby saving the time while taking an attendance. A modified

local binary pattern histogram algorithm is used it is mainly based on the pixel neighbourhood grey median for extracting main features of the student's face for producing more accurate result. This kind of Face recognition technology is much faster than the general attendance methods such as manual entry (or) swiping of Identity card [5].

G. Sittampalam and N. Ratnarajah. Here, the SAMS stands for Smart Attendance Management System which is an IOT based solution for managing the attendance in many top universities in which it uses both hardware and software components to start or automate the attendance process and improving the quality. The hardware component Smart Attendance Management System of consists of IOT-based devices such as RFID readers, biometric sensors which captures the attendance data in real-time. Hence the Students can use their RFID enabled Identity cards. And also, they can use the biometric features such as fingerprint, iris recognition and facial recognition to mark their attendance. The captured student's attendance data is transmitted to the cloud-based software platform for processing the further steps. The software component of Smart Attendance Management System provides a user-friendly interface to the users for managing attendance data which also generates the reports and monitors the attendance trends. This system also provides real-time alerts for attendance exceptions to the users such as late or absent for the students. Hence, the attendance data captured by Smart Attendance Management System can be analysed to identify patterns and trends, enabling universities to make data-driven decisions to improve student attendance management this kind of systems can also inhibit the interest in the students to attend the college regularly [6].

A. Hake, A. Samanta, P. Kasambe and R. Sutar. An Automatic Attendance Marker based on Beacon technology is also an IoTbased solution which utilizes the small Bluetooth Low Energy (BLE) devices known as beacons to track attendance in real-time. Beacons are the small, battery-powered devices that transmits the signal to the nearby smartphones or other Bluetooth enabled devices [7].

A. A. Sukmandhani and I. Sutedja. Face recognition is one of the popular methods used for online exams to confirm that the authenticity of the student taking the exam or not. It also involves using computer vision algorithms to identify the face of the student uniquely and match it with their registered photo ID to confirm their identity as true or fake. Before the exam starts, students are mandatorily required to register their photo ID after the registration then it is then stored securely in the database. During the exam of a student, the student's webcam is turned on, and then computer vision comes into the picture this algorithm starts capturing the

images of the student's face at regular interval of time. And then after images captured by the webcam are compared with the registered photo identity card (ID) using face recognition algorithm. The algorithm analyses facial features such as the shape of the face, nose, eyes, jaw-line mouth, and matches them with the photo ID of the student which was registered before the exam. If this algorithm determines that the student's face matches their registered photo ID, they are allowed to continue with the examination which is going on and proof for the presence of the students. Elsewhere If there is any mismatch, an alert will be triggered, and the student's exam is flagged for further review or he may quit the exam. The system also monitors any suspicious behaviour, such as sudden movements, multiple faces in the camera frame, which could indicate the alert that someone else is taking the exam and then the exam is flagged [8].

M. S. Mohd Azmi, M. H. Mohamed Zabil, K. C. Lim, R. F. Raja Azman, N. A. Noor Adnan and M. A. S. Mohamed Azman, "UNITEN Smart Attendance System (UniSas) Using Beacons Sensor," 2018 IEEE Conference on e-Learning, e-Management and e-Services (IC3e), Langkawi, Malaysia, 2018, pp. 35-39, doi: 10.1109/IC3e.2018.8632631. Smart attendance system is the new way of taking student attendance using smart phone devices together with intelligent sensors. In many universities in Malaysia, students' attendance need to be tracked and become one of the requirements for a student to pass a subject. Common approach of attendance taking is using paper based attendance. Besides that, RFID technology becoming commonly used to replace the paper based attendance taking method. In this study, a smart attendance system using Bluetooth low-energy (BLE) beacon (UniSas) is proposed and implemented in Universiti Tenaga Nasional (UNITEN). Further, the performance (accuracy, time-consumption and energy-consumption) of the paper-based attendance, RFID attendance system and UniSas are compared. Results showed that the proposed UniSas outperforms traditional method and any current smart attendance systems [9].

T. Adiono, D. Setiawan, Maurizfa, J. William and N. Sutisna, "Cloud Based User Interface Design for Smart Student Attendance System," 2021 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2021, pp. 1-5, doi: 10.1109/ISESD53023.2021.9501878. Current available attendance system cannot automatically analyze the student attendance data and its behavior. In order to solve this problem, a fingerprint based Smart Attendance System is designed. The system consists of IoT based fingerprint readers and cloud-based user interface. The fingerprint reader register and record student attendance. The cloud-based user interface consists of backend and frontend system. The backend store the data sent by fingerprint reader to the cloud storage, while the

frontend shows the data and its statistic. With this system, the abnormal student attendance behavior can be early detected. This paper presents the detail of cloud architecture and the user interface algorithm. The user interface has successfully show and detect the student attendance problem, statistic of student attendance, statistic of courses, statistic of room, and fingerprint reader status [10].

K. Navin, A. Shanthini and M. B. M. Krishnan, "A mobile based smart attendance system framework for tracking field personals using a novel QR code based technique," 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), Bengaluru, India, 2017, pp. 1540-1543, doi: 10.1109/SmartTechCon.2017.8358623. Tracking of a person for official purpose is a fiddly task. Quite a few scenarios that demands such requirements are tracking of attendance on a service person meeting a customer, sales person meeting a client or a health worker attending an outside patient to monitor adherence of treatment if they are frequently on field. It would require to monitor time of visit, the frequency and duration of contacts made by them, location of interaction are needed to be recorded and intimated to the concerned. To ensure better productivity and effectiveness in service in field tracking official work of field personals is a justifiable act. Constraint lies in implementing a automated attendance registering system which needs to be fool proof, cost effective, privacy protected and abiding ethical values. Many companies have made smart phones to be de facto requirement to be carried with them for employees like sales staff, service staff and health workers having field work. This paper proposes a smart phone based attendance framework which can overcome the above mentioned constraints. The proposed framework uses two different mobile applications used by field personals and recipient clients. The mobile apps requires simple internet connection and works on Quick Response(QR) code based technique as a mechanism for registering attendance remotely in to the web portal for field personals interacting respective recipient client. The high level implementation details of the proposed system is explained in this paper. It also discusses how the system verifies identity to eliminate false reporting [11].

A. Bejo, R. Winata and S. S. Kusumawardani, "Prototyping of Class-Attendance System Using Mifare 1K Smart Card and Raspberry Pi 3," 2018 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2018, pp. 1-5, doi: 10.1109/ISESD.2018.8605442**.** Currently, most of educational institutions in Indonesia use conventional mechanism for recording student's attendance in the class. The conventional system is based on signature filled in the paper-based attendance form. It suffers from several

problems such as time consuming, error prone and vulnerable to human error. To overcome those problems, in this paper, a prototype of online class-attendance system is developed. The proposed system is designed based on Mifare 1K smart card as student ID card and single board computer Raspberry Pi as the card reader as well as the gateway device. In order to operate the system, two software applications are introduced. One for card management tool called GM Card Manager and the other one for card reader tool called GM Card Attendance System. Experiments show that the proposed attendance system works well in term of creating, editing and formatting Mifare 1K student ID card and also recording the attendance data conveniently. The proposed attendance system perform faster data processing time, record more detail and accurate data, less error and less man power. This means that the proposed system is better than conventional system [12].

Z. Xu, P. Chen, W. Zhang, X. Liu and H. Wu, "Research on Mobile Phone Attendance Positioning System Based on Campus Network," 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), Xiangtan, China, 2019, pp. 387-389, doi: 10.1109/ICSGEA.2019.00094. With the promotion of "cooperative learning and research teaching" in colleges and universities, the demand for self-learning ability of students is getting higher and higher. How to supervise student's learning has become a problem that currently plagues colleges and universities. In order to better supervise and manage students, the attendance system plays an important role in the teaching and management of the school. With the popularity of computer networks, many colleges and universities have established campus networks. This makes it possible to check attendance for students in the network environment. At the same time, the popularity of smart mobile phones is getting higher and higher. Almost every student has a smart phone. Using the campus network and mobile phones to attend students' attendance has become a future development trend. This paper mainly studies the implementation method of mobile phone attendance and positioning system based on campus network in positioning and student verification [13].

R. V. Imbar, B. Renaldy Sutedja and M. Christianti, "Smart Attendance Recording System using RFID and e-Certificate using QR Code-based Digital Signature," 2021 International Conference on ICT for Smart Society (ICISS), Bandung, Indonesia, 2021, pp. 1-5, doi: 10.1109/ICISS53185.2021.9533199. Nowadays, the development of Information Technology is growing very rapidly. The term smart is widely used in the field of Information Technology as a smart system such as a smart library, smart campus. There are many components of technology associated with smart campuses, one of which is the Internet of Things. RFID is

one of the Internet of Things solutions that will have a positive impact on universities. This paper will discuss how to implement a smart attendance recording system using RFID and integrate the e-Certificate system for organizing activities at Maranatha Christian University, by a QR Code-based digital signature as a guarantee of the authenticity of the e-Certificate generated [14].

R. Vijayakumar, M. Poornima, S. Divyapriya and T. Selvaganapathi, "Automated Student Attendance Tracker for End Semester Examination using Face Recognition System," 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2022, pp. 1566-1570, doi: 10.1109/ICOSEC54921.2022.9952035. The keeping and recording of a class attendance log manually is not very efficient for end semester examination hall tickets generation due to maintaining minimum attendance percentage for each subjects. Since offering proxies for absentees or skipping class has become amusing and a fantasy among students of the current age. Manually recording attendance in logbooks becomes challenging and vulnerable to manipulation. Therefore, the purpose of this project is to present the automatic attendance system. This system recognizes each student's face to automatically identify them in the classroom and record their attendance and also record wherever roaming inside the campus. By photographing students' faces in real time, this system was created. The reference faces in the dataset are compared to the detected faces, marking the attendees' attendance. At the end of the semester all the student’s attendances are calculated automatically and give notification to students, parents and class advisors for the status of attendance percentage. Based on the detailed results, hall tickets are automatically generate in the college exam cell. The final project is completed and reports were generated [15].

S. Anand, K. Bijlani, S. Suresh and P. Praphul, "Attendance Monitoring in Classroom Using Smartphone & Wi-Fi Fingerprinting," 2016 IEEE Eighth International Conference on Technology for Education (T4E), Mumbai, India, 2016, pp. 62-67, doi: 10.1109/T4E.2016.021. Academic performance is directly affected by student attendance during the lecture hours. There are existing manual and automated attendance tracking systems that work to ensure that students attend the lectures without fail. However, the practical implementation of most automated systems have drawbacks such as high monetary cost, the need to install specialized hardware, and proneness to fake or proxy attendance. To address this, we propose a novel attendance marking system with which students may mark attendance using their smartphones. While applying facial recognition via the smartphone's front camera to determine the student's identity, the system also makes use of the campus Wi-Fi network to determine the student’s

location. The proposed system does not require high monetary cost or specialized hardware and yet incorporates adequate foolproof measures to counter fake or proxy attendance. Experimental studies with our system show that fingerprinting, which is the technique used here to determine indoor location, can achieve very good positioning accuracy even in classroom environments, where signal interference is usually very high [16].

S. K. Baharin, Z. Zulkifli and S. B. Ahmad, "Student Absenteeism Monitoring System Using Bluetooth Smart Location-Based Technique," 2020 International Conference on Computational Intelligence (ICCI), Bandar Seri Iskandar, Malaysia, 2020, pp. 109-114, doi: 10.1109/ICCI51257.2020.9247809. Conventional method of recording students' attendance is still being used in Universiti Teknologi MARA (UiTM) Perak Tapah Campus. The method used is by recording students' attendance in attendance sheets which is an inefficient way to monitoring students' attendance. The absenteeism of students without valid excuses during lectures appears to be a serious problem as it falls under the term of truancy. Once the absenteeism percentage reaches 10%, the students will receive a notification letter issued by the Academic Affairs Division as the first warning. The last warning will be issued when the absenteeism percentage reaches 20% where the students might be barred from sitting the final examination. Therefore, the "Student Absenteeism Monitoring System Using Bluetooth Smart Location-Based Technique" is developed specifically for lecturers and students of the UiTM Tapah Campus to automatically monitor students' attendance. The objective of this project is to determine the percentage of students' absenteeism to prevent students from getting a ZZ status. ZZ status is a situation where a student is being barred from sitting the final examination. The system was evaluated based on functionality, usability efficiency, and user acceptance test. The result from evaluations indicates that most of the users have good experience in using the system. This system which was specifically developed for UiTM can also be enhanced and customized to meet the needs of learning institutions throughout Malaysia [17].

A. Shene, J. Aldridge and H. Alamleh, "Privacy-Preserving Zero-effort Class Attendance Tracking System," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2021, pp. 1-4, doi: 10.1109/IEMTRONICS52119.2021.9422481. Student attendance tracking is a vital process in education. This process can be tiring and time-consuming. We believe it is possible to automate it using technologies available in the educational infrastructure and user smartphones. Today smartphones can sense several types of signals over the air using radio frequency technologies (e.g., Wi-Fi, Bluetooth, cellular signals, etc.). Furthermore, smartphones receive broadcast

messages from transmitting entities and can measure the received signal strength. We believe that these signals can be utilized in the context of classroom attendance tracking, primarily because they can indicate the location of a user's device. The proposed system aims to have student smartphones in the classroom generate “location proofs” based on the radio frequency fingerprints scanned by their devices, which are later used to verify their locations. In this paper, we propose the utilization of Wi-Fi access points in buildings on school campuses in conjunction with the instructor and student smartphones to build a zero-effort and privacy-preserving attendance tracking system. Our system is unique as it does not require any effort from users in the system. Moreover, it is privacy-preserving, as the App server has no information about user identities nor class locations [18].

M. Hu and H. Li, "Application of Location-Controlled Mobile Attendance Recording System in College Classroom Teaching," 2020 International Symposium on Educational Technology (ISET), Bangkok, Thailand, 2020, pp. 18-22, doi: 10.1109/ISET49818.2020.00014. In the process of college classroom teaching activities, it is necessary to record the attendance of the students. The traditional methods are through the manual way naming statistics, need to take up a lot of time, affecting the normal classroom teaching activities. For the social practice classes, online classrooms, large-scale lectures and other such teaching activities, traditional attendance records methods apparently can't meet the education requirements. In this paper, a mobile attendance record system based position controllable classroom instruction is presented. Students will submit the attendance and location information through the attendance record APP. The teachers can set the attendance location to control the learners must be to the specified location for attendance records or remote attendance records. The experimental results showed that the system is flexible and straightforward, which can save attendance and statistic time, improve teaching efficiency and meet the needs of attendance record in scene teaching activities [19].

S. Sharma, S. Monika, S. V. S. Prasad, K. Dasari and S. Kamaganikuntla, "RFID Based Low Cost Attendance Recording and Proxy Avoidance System," 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), Bhubaneswar, India, 2022, pp. 1-5, doi: 10.1109/ASSIC55218.2022.10088295. In educational institutes, taking attendance on each day is mandatory. The conventional way of taking attendance is by calling every individual with their names or by signing on some sheet. This method might cause some errors and consume more time, which makes this method inefficient. Giving proxy to other student is one of the major drawbacks in the traditional way. So, to avoid all this, we came up

with RFID (Radio Frequency Identification) based low-cost attendance recording system. In this system, we use RFID technology to take attendance, where each student is issued with an RFID tag. Each tag is contained with some unique information about the student. When the tag is placed near the reader, the reader reads the information from the tag and sends it to the Arduino board. The controller checks for the data and compares with the data base. If the tag is valid, the controller marks the student as present and opens the classroom door for entry. The main components of the proposed prototype hardware setup are Arduino uno, RFID tags, tag reader, servo motor, IR (Infra-Red) sensor and LCD (Liquid Crystal Display) [20].

V. Mishra, S. Raj, T. Singhal and C. Sankhla, "Intelligent Face Recognition based attendance system," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-4, doi: 10.1109/ICRITO56286.2022.9964862. Student consistent performance is a challenge for educational institutions worldwide, especially in India. Inadequate attendance is one of the prime reasons that is often correlated with this drop in student performance. The method generally employed in academic institutions to register attendance is the manual record method by either signing or calling out the pupil's name. This takes a long time and is inefficient, e.g., missing out on names and proxy attendances. Professors are increasingly relying on a computer-based student attendance verification system to help them keep track of attendance. Our project deploys a facial recognition-based attendance system. Automated facial recognition systems have made tremendous improvements in today's environment. We found out that Real-Time Face Recognition is an appropriate system for recording and tracking daily attendance. This type of attendance system involves recording the attendance using high-definition (HD) monitor video and related technologies for facial recognition of students. Our facial recognition system does this by analyzing the image captured by the security cameras. Although we employed a range of algorithms and software to achieve this, the primary concept to be used here is Deep Learning. It facilitates the conversion of video frames to photographs, allowing the student's face to be easily recognized for attendance purposes and the attendance database to be automatically updated [21].

V. Mishra, S. Raj, T. Singhal and C. Sankhla, "Intelligent Face Recognition based attendance system," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-4, doi: 10.1109/ICRITO56286.2022.9964862. Student consistent performance is a challenge for educational institutions worldwide, especially in India. Inadequate attendance is one of the

prime reasons that is often correlated with this drop in student performance. The method generally employed in academic institutions to register attendance is the manual record method by either signing or calling out the pupil's name. This takes a long time and is inefficient, e.g., missing out on names and proxy attendances. Professors are increasingly relying on a computer-based student attendance verification system to help them keep track of attendance. Our project deploys a facial recognition-based attendance system. Automated facial recognition systems have made tremendous improvements in today's environment. We found out that Real-Time Face Recognition is an appropriate system for recording and tracking daily attendance. This type of attendance system involves recording the attendance using high-definition (HD) monitor video and related technologies for facial recognition of students. Our facial recognition system does this by analyzing the image captured by the security cameras. Although we employed a range of algorithms and software to achieve this, the primary concept to be used here is Deep Learning. It facilitates the conversion of video frames to photographs, allowing the student's face to be easily recognized for attendance purposes and the attendance database to be automatically updated [22].

P. Gupta and B. Singh, "A New Way of Recording Attendance of the Students using Face Recognition System," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, 2022, pp. 578-583, doi: 10.1109/IC3I56241.2022.10073382. People’s connections at work affect the environment there, which is a crucial aspect of everyday supervision. However, keeping track of people’s present and absences may be done via attendance. For many years, techniques that need paper have been used to record attendance of the students. This engagement strategy had been around for a while. For a small number of participants, this approach could be suitable and workable, but it would take a lot of time and effort for a larger group. This approach has well-known and quite well disadvantages, including time requirements, an error-prone nature, and the potential for proxy participation. Using facial recognition software, we provide a useful and stylish solution to track attendance. In order to ascertain if a person is there, the technology will recognize their face and comparing it to the database. Additionally, the vital information is retained with school records and time, and if required, manual changes may be made. This method may be used to address the issue of intermediaries and false attendees [23].

D. Yadav, S. Maniar, K. Sukhani and K. Devadkar, "In-Browser Attendance System using Face Recognition and Serverless Edge Computing," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2021,

pp. 01-06, doi: 10.1109/ICCCNT51525.2021.9580042. Because of Covid-19, schools, colleges, and institutions have moved to online learning. The education system has encountered and continues to encounter various challenges in this online format in managing the attendance of the students. The teacher used to call out the students' roll numbers or names when they were in the physical education mode. Nowadays as the world is developing towards a digital era, numerous techniques of collecting attendance such as attendance via biometric technologies like eye recognition, face scanning, voice recognition, fingerprint analysis have earned a lot of fame. Face recognition is the most efficient of these approaches as the face can be captured using a camera and compared using a trained model, but the others are more complex to implement at the user end, and some even need hardware. A lot of research work has been already done related to face recognition using models such as YOLO, MTCNN, FaceNet, HOG, LBPH, C2D-CNN. Models are usually loaded in the backend which causes latency issues and makes the system inefficient to use. Our proposed system aims to perform face recognition within the browser itself with the help of serverless edge computing. For the students, a simple web portal is developed, from which they can navigate to our plugin extension, where the model will capture attendance and dynamically update it in a Google Sheet. Face detection was done with Tiny Face Detector, while face recognition was done with Face Recognition Net. A few more models operate in conjunction with these two, recognizing the student from his or her livestream, checking the student's authenticity using logged in credentials, and updating the attendance in real-time across the browser [24].

Efanntyo and A. R. Mitra, "Masked Face Recognition by Applying SSD and ResNet Model for Attendance System," 2021 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA), Surabaya, Indonesia, 2021, pp. 234-238, doi: 10.1109/ICAMIMIA54022.2021.9807814. The rapid development of technology in artificial intelligence (AI) includes face recognition. Some of these developments in face recognition systems can be found easily in everyday life, such as entry access, payment, or attendance recording systems. Notably, the demand for image-based face recognition methods to record attendance increases due to its effectiveness and efficiency. Compared to conventional methods like fingerprint or RFID, facial recognition offers better results. On the other hand, with the arising needs in dealing with the novel coronavirus (COVID-19) pandemic, official health protocols require mask-wearing and maintaining a minimum distance of 1 meter between individuals to prevent the spread of the virus. Along with such a situation, a face recognition system to record attendance can reduce the occurrence of direct contact and allow

each individual to maintain a safe distance, including from the attendance device. This paper presents the performance of a masked-faced recognition system that implemented SSD (Single Shot Detection) and ResNet feature extraction. The face recognition system application developed using Python and related libraries show a stable level of masked face recognition accuracy. This evaluation was made at predetermined distances between the face and the camera and measured at room lighting of 200 lux with an average accuracy of 67%. The application also has a feature to send notification emails to every employee who is unable to attend work on their scheduled workdays [25].

CHAPTER 3

EXISTING SOLUTION & METHODOLOGY

3.1 EXISTING SOLUTION

Contrary to our proposed system, there are many existing solutions to the problem that was mentioned above. They use different technologies and methods for a better attendance management. Some of them are below.

a) Using PCA algorithm

In this system, they have used a mobile phone camera along with machine learning algorithm (PCA), which is commonly used for face detection and low-end face recognition. The overall price of the system is comparatively less and very portable to use. Students can mark their attendance while using their own mobile phone camera.

b) RFID Technology

The method to use RFID technology is very common and it is mostly used in shopping malls etc., But here, the student id cards function as a RFID tag and using a RFID reader, they can scan their cards to mark their attendance. It is very simple to use.

c) Beacon Technology

A Bluetooth beacon is used, which has the radius of 2-3 meters. It detects the presence of the students through their own smartphone Bluetooth, which has a unique ID. The student needs to be present in the allocated range for his attendance to be marked as present.

d) Minutiae based Algorithm

In this, the biometrics like fingerprint are taken as a method to mark attendance. The students need to scan their fingerprint after entering their unique id, then the device will match the fingerprint data to check for the similarities along the ridge pattern. The Minutiae algorithm is used for this as its coverts the 2D scanned data into a 3D model for analyzing.

e) LBPH Algorithm

The algorithm uses face liveliness to detect and recognize faces. It takes around 200 images of each student for a better recognition. It first finds faces, posing or working faces, differentiates them, encode them, and find the person encoded through database. But the system is

implemented only for online use, where for classes, using the laptop camera, the attendance of students is marked.

Through our proposed system we tried to solve most of the cons in the current existing solution. It is relatively cheaper, fast and secure. Using CNN algorithm, it detects and recognizes faces in real time while also maintaining an accuracy rate of more than 90%. The reliability is more and as we have made a physical module, it can be directly used in the class. It is better at crowd sensing and works just with one image for each person. CNN algorithm separates a single image into multiple layers and using HAAR classification, it segregates positive and negative data points to be able to better detect the student's face.

3.2 METHODOLOGY

An Attendance management system is developed, which uses face recognition as its primary method to mark attendance of a student on a daily basis. It is a convenient method, developed to replace the traditional method of taking attendance in a class. At the end, the system provides a list of data of students that are marked present in the .csv format. The main components used are raspberry pi and a portable camera for its real time image processing, interfacing and IOT connectivity. The user related work is done here which also includes taking inputs or converting given data sets. Later, at the backend, firebase, an open-source cloud-based web application platform is used, which acts as a database for the system.

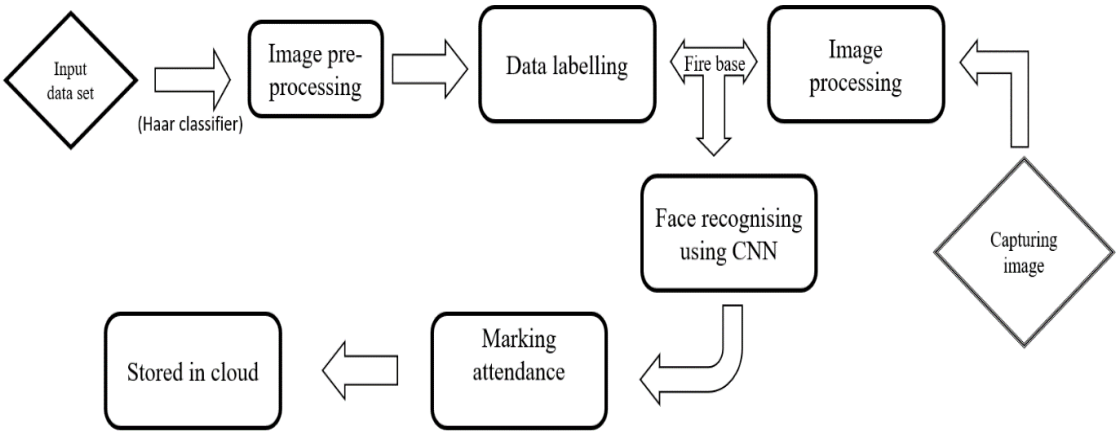


Figure 3.2.1 Functional diagram of attendance management system.

As we could imagine, in the tradition method of taking attendance, for a student, there is chance to miss the roll call. It also effects his daily evaluation, where the fault lies in the method of taking attendance. So, a system where it is replaced with a more accurate and less labor involving is proposed here. It also helps for time consistency of a teacher by allowing them to utilize an allocated period of time just for teaching and nothing else.

We feed in all the data sets, where, a set of facial data of the student is collected through an external camera model. Each image covers the entire face from the front for a better detection and more accurate results in real time. Each image is of resolution 1024*768 and is converted to 786432*1 and stored in the firebase, an open-source cloud-based application platform. Using CNN algorithm, the attributes of each face is converted into 128 mathematical numbers and

stored in the database. While take the live input, the image is processed through multiple layers in CNN to compare the stored data and the input data as shown below.

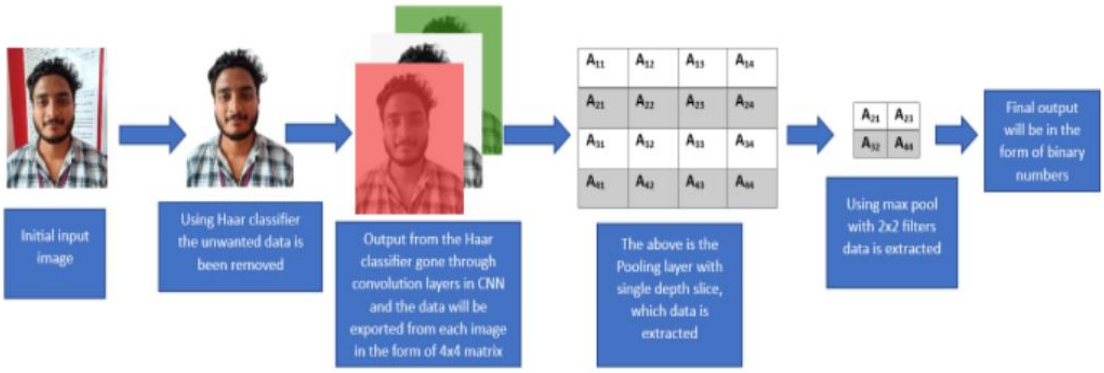


Figure 3.2.2 An Overview of the complete working process.

With an external pi 3b camera module, the face of the student is captured in real time, when he enters the classroom and the input is immediately stored in the local database with a .rasp file name. The face data matching is done by comparing the input data and the stored data. If the face attributes of given data are matched with an pre-stored dataset, then it gives out the output of the student name it is matched with and stores it in excel sheet format in the cloud. It becomes accessible to people with right credentials, usually, a teacher or as student.

3.3 ALGORITHMS

CONVOLUTIONAL NEURAL NETWORK

Computer vision is evolving rapidly day-by-day. Its one of the reason is deep learning. When we talk about computer vision, a term convolutional neural network(abbreviated as CNN) comes in our mind because CNN is heavily used here. Examples of CNN in computer vision are face recognition, image classification etc. It is similar to the basic neural network. CNN also have learnable parameter like neural network i.e, weights, biases etc.

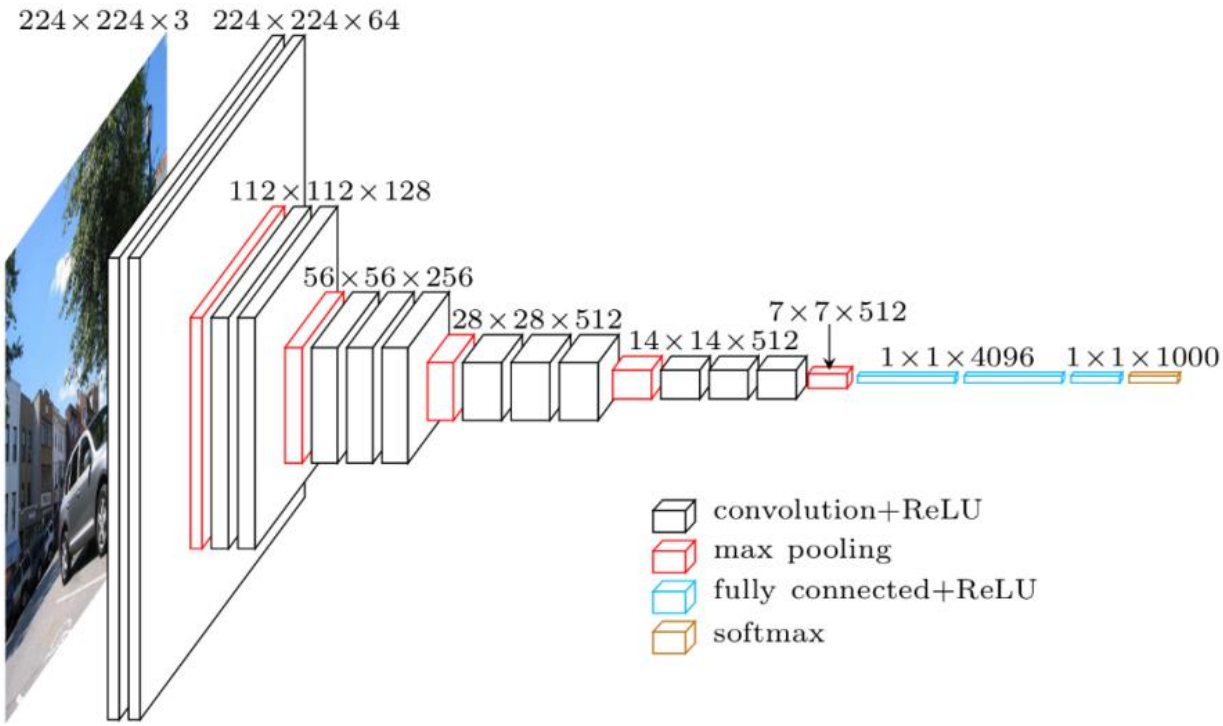


Figure 3.3.1 Downsampling.

Suppose you are working with MNIST dataset, you know each image in MNIST is 28 x 28 x 1(black & white image contains only 1 channel). Total number of neurons in input layer will 28 x 28 = 784, this can be manageable. What if the size of image is 1000 x 1000 which means you need 10^6 neurons in input layer. Oh! This seems a huge number of neurons are required for operation. It is computationally ineffective right. So here comes Convolutional Neural Network or CNN. In simple word what CNN does is, it extract the feature of image and convert it into lower dimension without loosing its characteristics. In the following example you can see that initial the size of the image is 224 x 224 x 3. If you proceed without convolution then you need 224 x 224 x 3 = 100, 352 numbers of neurons in input layer but after applying

convolution you input tensor dimension is reduced to 1 x 1 x 1000. It means you only need 1000 neurons in first layer of feedforward neural network.

IMAGE REPRESENTATION

Thinking about images, its easy to understand that it has a height and width, so it would make sense to represent the information contained in it with a two dimensional structure (a matrix) until you remember that images have colors, and to add information about the colors, we need another dimension, and that is when Tensors become particularly helpful.

Images are encoded into color channels, the image data is represented into each color intensity in a color channel at a given point, the most common one being RGB, which means Red, Blue and Green. The information contained into an image is the intensity of each channel color into the width and height of the image, just like this

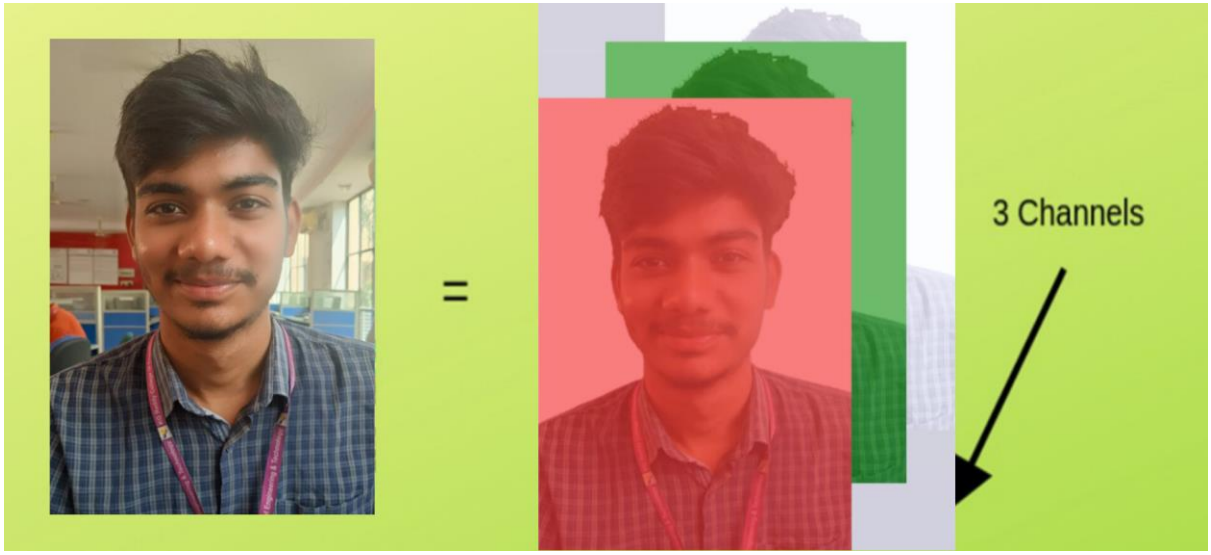


Figure 3.3.2 RGB representation of a image.

So the intensity of the red channel at each point with width and height can be represented into a matrix, the same goes for the blue and green channels, so we end up having three matrices, and when these are combined they form a tensor.

Edge Detection

Every image has vertical and horizontal edges which actually combining to form a image. Convolution operation is used with some filters for detecting edges. Suppose you have gray scale image with dimension 6 x 6 and filter of dimension 3 x 3(say). When 6 x 6 grey scale image convolve with 3 x 3 filter, we get 4 x 4 image. First of all 3 x 3 filter matrix get multiplied

with first 3 x 3 size of our grey scale image, then we shift one column right up to end , after that we shift one row and so on.

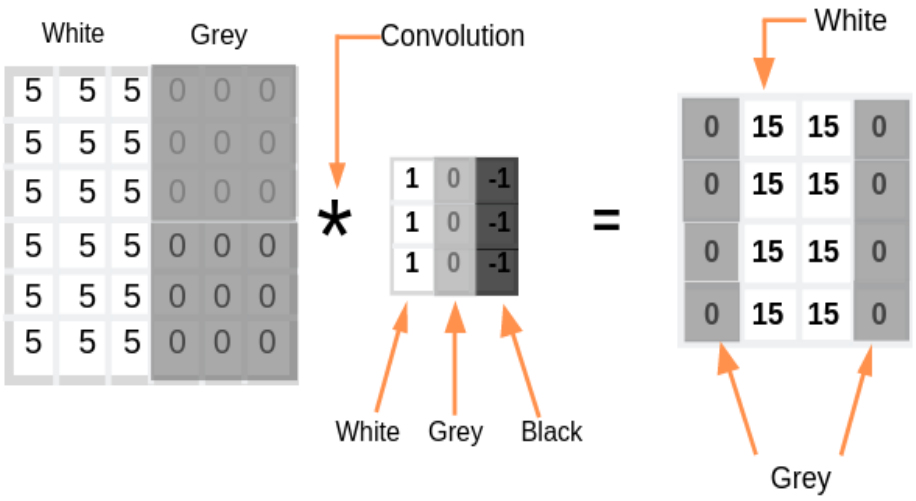


Figure 3.3.3 Convolutional operation.

The convolution operation can be visualized in the following way. Here our image dimension is 4 x 4 and filter is 3 x 3, hence we are getting output after convolution is 2 x 2.

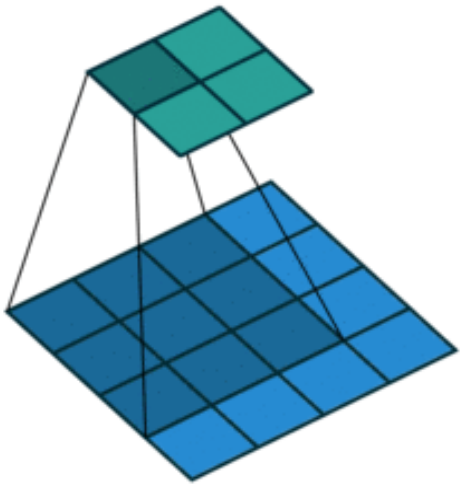


Figure 3.3.4 Visualization of Convolutional.

If we have N x N image size and F x F filter size then after convolution result will be

```
(N x N) * (F x F) = (N-F+1)x(N-F+1)(Apply this for above case)
```

Stride and Padding

Stride denotes how many steps we are moving in each steps in convolution.By default it is one.

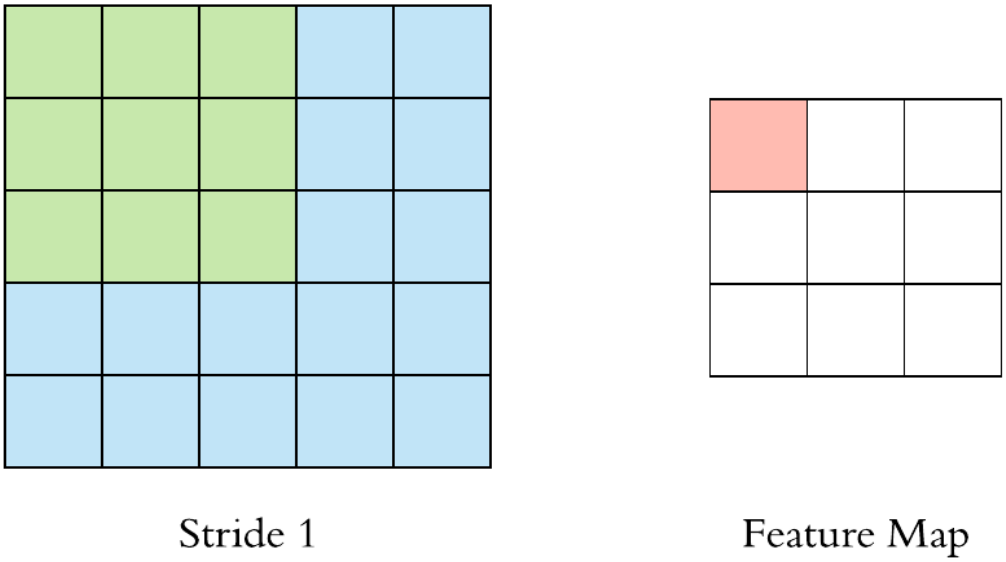


Figure 3.3.5 Convolutional with Stride 1.

We can observe that the size of output is smaller that input. To maintain the dimension of output as in input , we use padding. Padding is a process of adding zeros to the input matrix symmetrically. In the following example,the extra grey blocks denote the padding. It is used to make the dimension of output same as input.

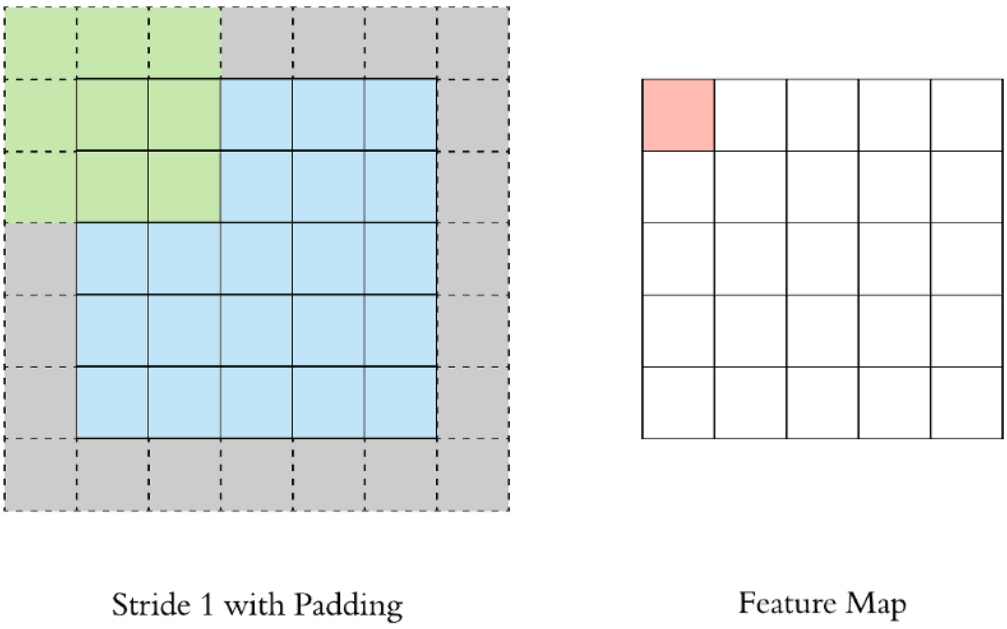


Figure 3.3.6 Stride 1 with padding 1.

Let say ‘p’ is the padding

Initially(without padding)

```
(N x N) * (F x F) = (N-F+1)x(N-F+1)---(1)
```

After applying padding

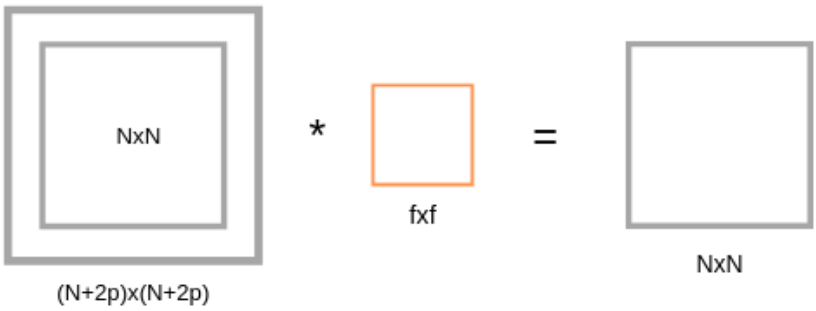


Figure 3.3.7 After applying padding.

If we apply filter F x F in (N+2p) x (N+2p) input matrix with padding, then we will get output matrix dimension (N+2p-F+1) x (N+2p-F+1). As we know that after applying padding we will get the same dimension as original input dimension (N x N). Hence we have,

```
(N+2p-F+1)x(N+2p-F+1) equivalent to NxN
 N+2p-F+1 = N ---(2)
 p = (F-1)/2 ---(3)
```

The equation (3) clearly shows that Padding depends on the dimension of filter.

Layers in CNN

There are five different layers in CNN

- Input layer
- Convo layer (Convo + ReLU)
- Pooling layer
- Fully connected(FC) layer

- Softmax / logistic layer
- Output layer

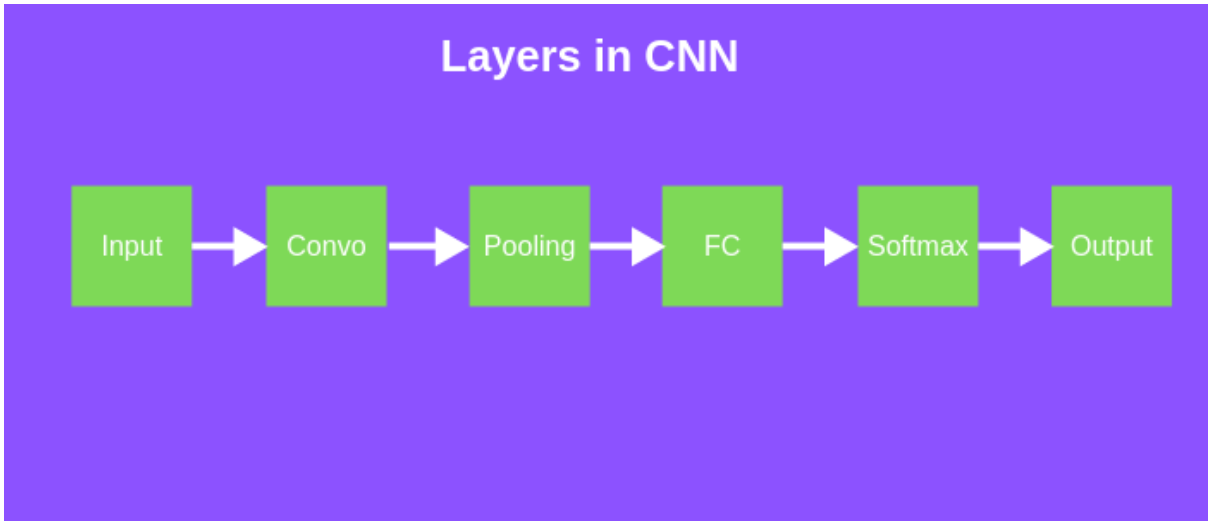


Figure 3.3.8 Different layers in CNN.

Input Layer

Input layer in CNN should contain image data. Image data is represented by three dimensional matrix as we saw earlier. You need to reshape it into a single column. Suppose you have image of dimension 28 x 28 =784, you need to convert it into 784 x 1 before feeding into input. If you have "m" training examples then dimension of input will be (784, m).

Convo Layer

Convo layer is sometimes called feature extractor layer because features of the image are get extracted within this layer. First of all, a part of image is connected to Convo layer to perform convolution operation as we saw earlier and calculating the dot product between receptive field(it is a local region of the input image that has the same size as that of filter) and the filter. Result of the operation is single integer of the output volume. Then we slide the filter over the next receptive field of the same input image by a Stride and do the same operation again. We will repeat the same process again and again until we go through the whole image. The output will be the input for the next layer.

Convo layer also contains ReLU activation to make all negative value to zero

Pooling Layer

Pooling layer is used to reduce the spatial volume of input image after convolution. It is used between two convolution layer. If we apply FC after Convo layer without applying pooling or max pooling, then it will be computationally expensive and we don't want it. So, the max pooling is only way to reduce the spatial volume of input image. In the above example, we have applied max pooling in single depth slice with Stride of 2. You can observe the 4 x 4 dimension input is reduce to 2 x 2 dimension.

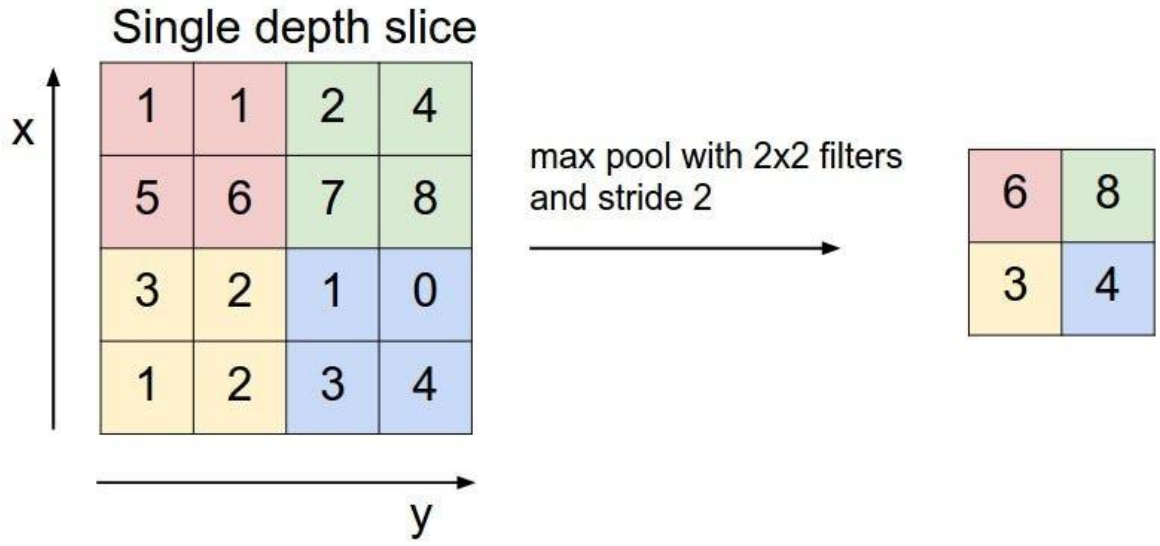


Figure 3.3.9 CS231 n Convolutional Neural Networks.

There is no parameter in pooling layer but it has two hyperparameters — Filter(F) and Stride(S).

In general, if we have input dimension W1 x H1 x D1, then

$W2 = (W1-F)/S+1$

$H2 = (H1-F)/S+1$

$D2 = D1$

Where W2, H2 and D2 are the width, height and depth of output.

Fully Connected Layer (FC)

Fully connected layer involves weights, biases, and neurons. It connects neurons in one layer to neurons in another layer. It is used to classify images between different category by training.

Softmax / Logistic Layer

Softmax or Logistic layer is the last layer of CNN. It resides at the end of FC layer. Logistic is used for binary classification and softmax is for multi-classification.

Output Layer

Output layer contains the label which is in the form of one-hot encoded. Now you have a good understanding of CNN. Let's implement a CNN in Keras

Keras Implementation

We will use CIFAR-10 dataset to build a CNN image classifier. CIFAR-10 dataset has 10 different labels

- Airplane
- Automobile
- Bird
- Cat
- Deer
- Dog
- Frog
- Horse
- Ship
- Truck

It has 50,000 training data and 10,000 testing image data. Image size in CIFAR-10 is 32 x 32 x 3. It comes with Keras library.

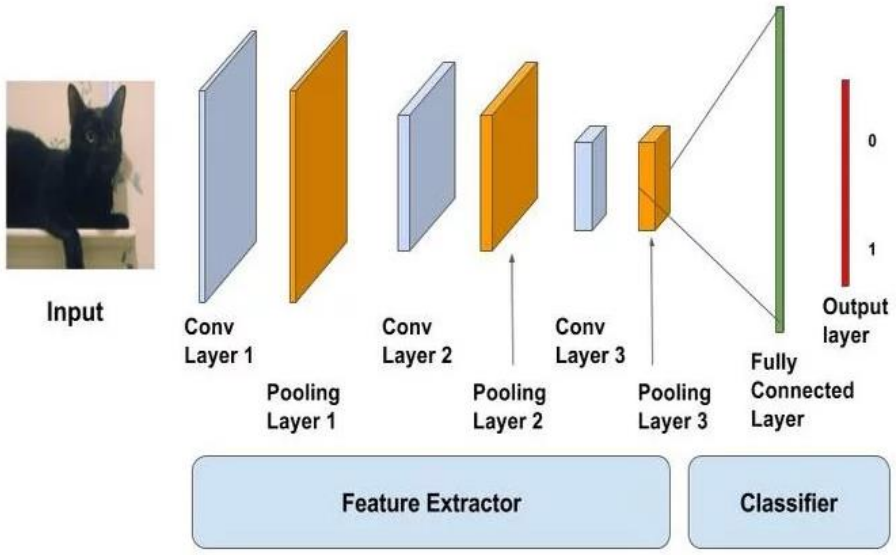


Figure 3.3.10 Model Visualization.

HAAR CASCADE CLASSIFIER

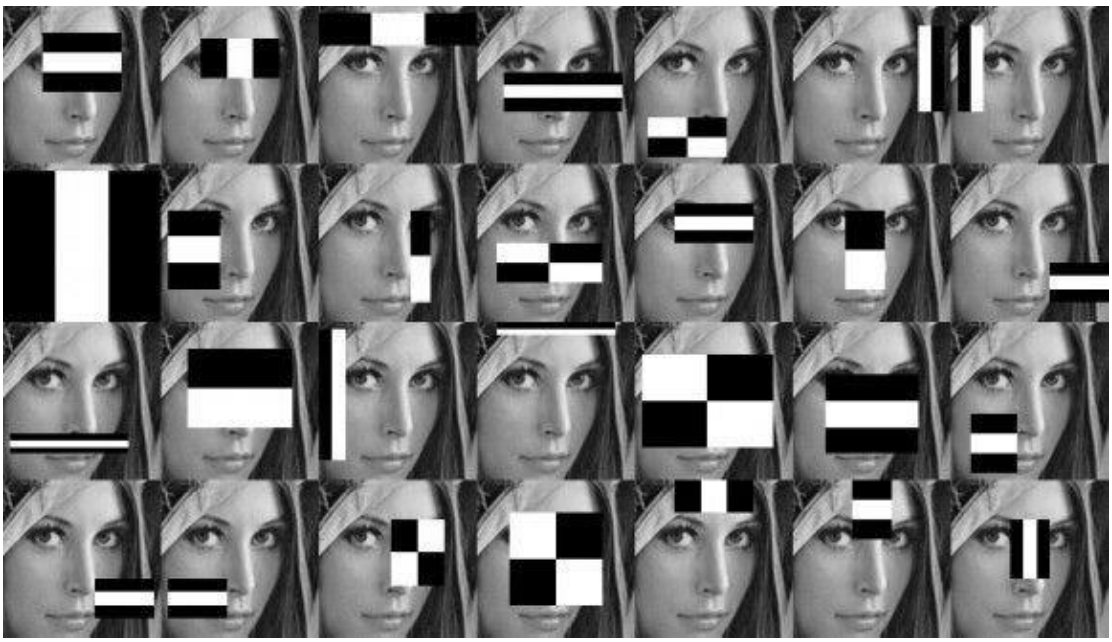


Figure 3.3.11 A general representation of training a Haar Classifier.

Haar Cascade classifiers are an effective way for object detection. This method was proposed by Paul Viola and Michael Jones in their paper Rapid Object Detection using a Boosted Cascade of Simple Features.Haar Cascade is a machine learning-based approach where a lot of positive and negative images are used to train the classifier.

Making a Haar Cascade Classifier

The algorithm can be explained in four stages:

- Calculating Haar Features
- Creating Integral Images
- Using Adaboost
- Implementing Cascading Classifiers

It's important to remember that this algorithm requires a lot of **positive images** of faces and **negative images** of non-faces to train the classifier, similar to other machine learning models.

Calculating Haar Features

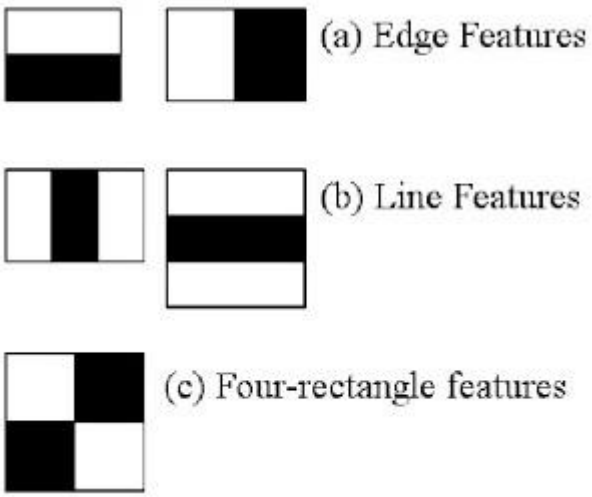


Figure 3.3.12 Types of Haar features.

The first step is to collect the Haar features. A Haar feature is essentially calculations that are performed on adjacent rectangular regions at a specific location in a detection window. The calculation involves summing the pixel intensities in each region and calculating the differences between the sums. Here are some examples of Haar features below.

These features can be difficult to determine for a large image. This is where integral images come into play because the number of operations is reduced using the integral image.

Creating Integral Images

Without going into too much of the mathematics behind it (check out the paper if you're interested in that), integral images essentially speed up the calculation of these Haar features. Instead of computing at every pixel, it instead creates sub-rectangles and creates array references for each of those sub-rectangles. These are then used to compute the Haar features.

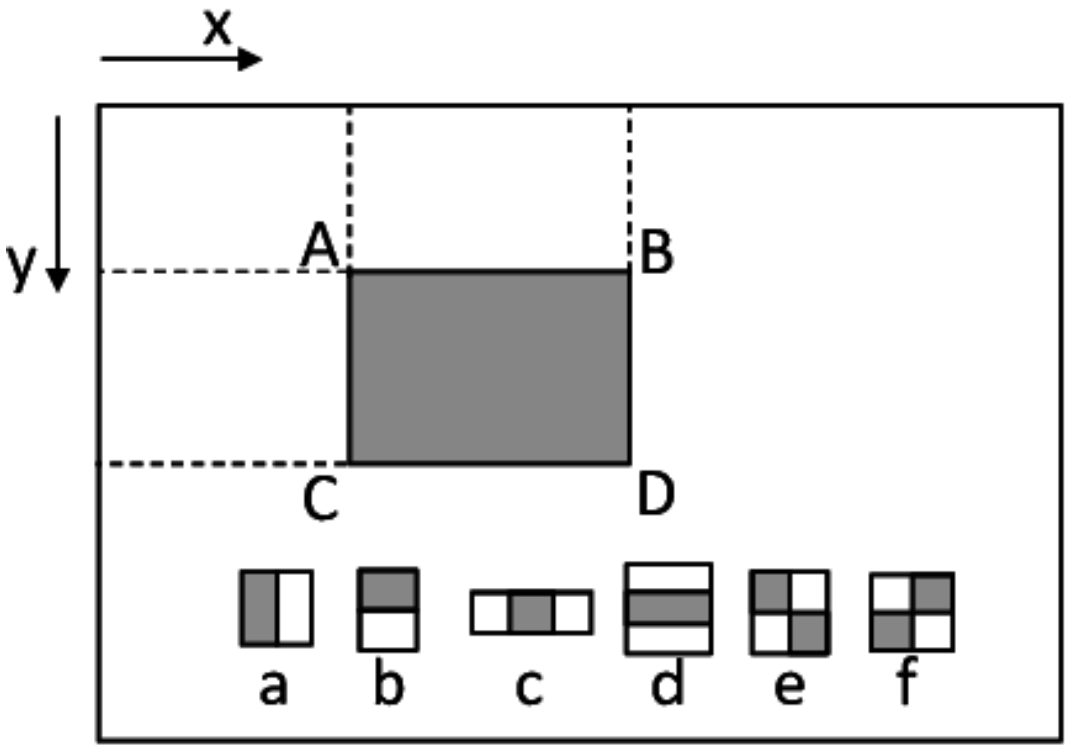


Figure 3.3.13 Illustration for how an internal image works.

It's important to note that nearly all of the Haar features will be irrelevant when doing object detection, because the only features that are important are those of the object. However, how do we determine the best features that represent an object from the hundreds of thousands of Haar features? This is where Adaboost comes into play.

Adaboost Training

Adaboost essentially chooses the best features and trains the classifiers to use them. It uses a combination of **"weak classifiers"** to create a **"strong classifier"** that the algorithm can use to detect objects.

Weak learners are created by moving a window over the input image, and computing Haar features for each subsection of the image. This difference is compared to a learned threshold that separates non-objects from objects. Because these are "weak classifiers," a large number of Haar features is needed for accuracy to form a strong classifier.

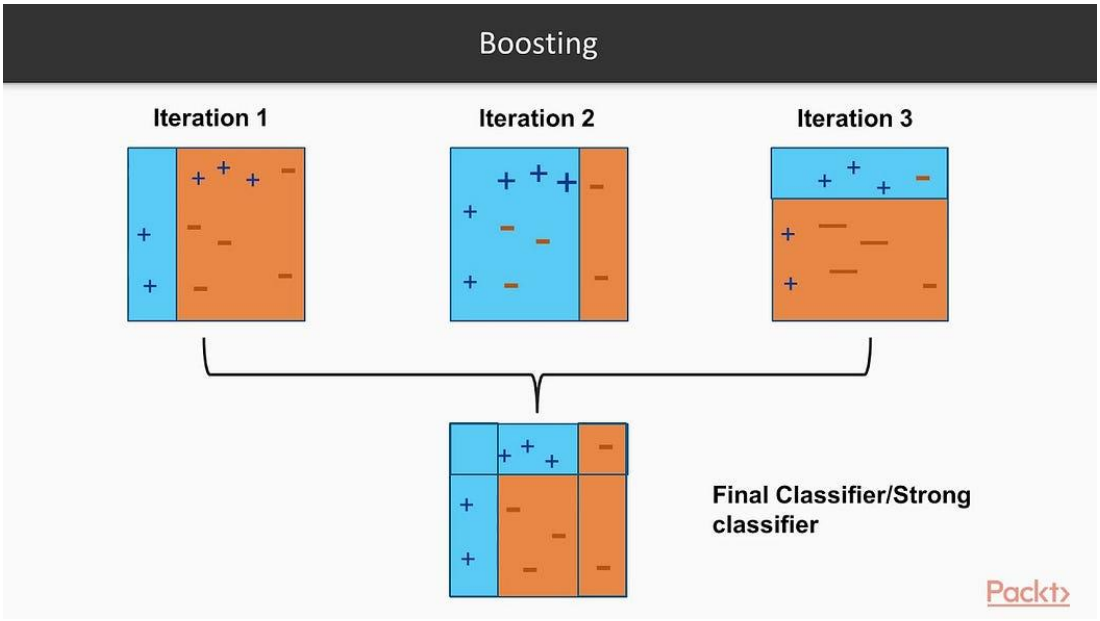


Figure 3.3.14 Representation of a boosting algorithm.

The last step combines these weak learners into a strong learner using cascading classifiers.

Implementing Cascading Classifiers

The cascade classifier is made up of a series of stages, where each stage is a collection of weak learners. Weak learners are trained using boosting, which allows for a highly accurate classifier from the mean prediction of all weak learners.

Based on this prediction, the classifier either decides to indicate an object was found (positive) or move on to the next region (negative). Stages are designed to reject negative samples as fast as possible, because a majority of the windows do not contain anything of interest.

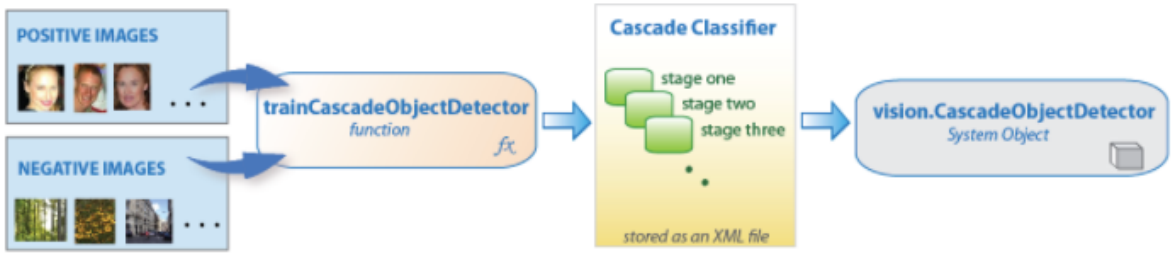


Figure 3.3.15 A flowchart of cascade classifiers.

It's important to maximize a low false negative rate, because classifying an object as a non-object will severely impair your object detection algorithm. A video below shows Haar cascades in action. The red boxes denote "positives" from the weak learners.

CHAPTER 4

HARDWARE AND SOFTWARE DESCRIPTION

4.1 HARDWARE SPECIFICATIONS

4.1.1: Raspberry Pi

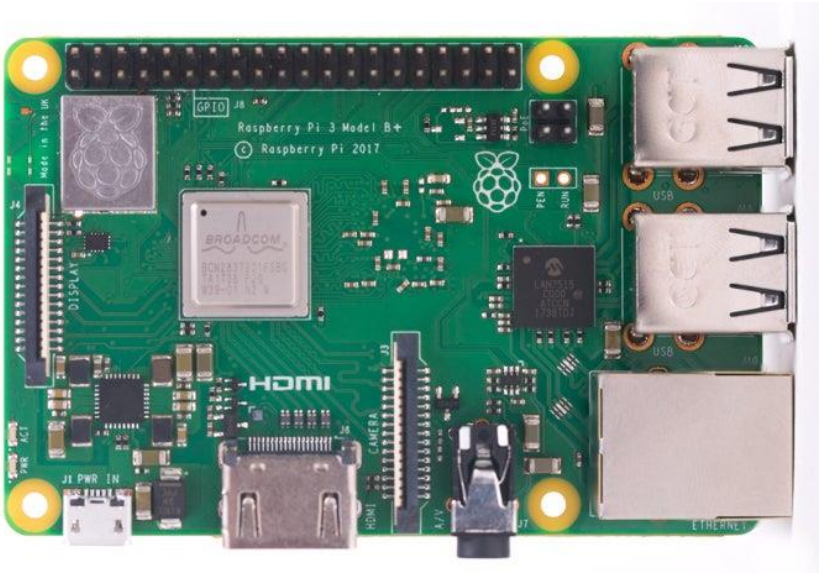


Figure 4.1.1 Raspberry Pi B+ model.

Raspberry Pi is a series of small single-board computers (SBCs) developed in the United Kingdom by the Raspberry Pi Foundation in association with Broadcom. The Raspberry Pi project originally leaned towards the promotion of teaching basic computer science in schools. The original model became more popular than anticipated, selling outside its target market for uses such as robitics. It is widely used in many areas, such as for, because of its low cost, modularity, and open design. It is typically used by computer and electronic hobbyists, due to its adoption of the DDMI and USB standards.

After the release of the second board type, the Raspberry Pi Foundation set up a new entity, named Raspberry Pi Trading, and installed Eben Upton as CEO, with the responsibility of developing technology.[20] The Foundation was rededicated as an educational charity for promoting the teaching of basic computer science in schools and developing countries. Most Pis are made in a Sony factory in Pencoed, Wales,[21] while others are made in China and Japan.

In 2015 the Raspberry Pi surpassed the ZX Spectrum in unit sales, becoming the best selling British computer.

Series and generations

There are three series of Raspberry Pi, and several generations of each have been released. Raspberry Pi SBCs feature a Broadcom system on a chip (SoC) with an integrated ARM-compatible central processing unit (CPU) and on-chip graphics processing unit (GPU), while Raspberry Pi Pico has a RP2040 system on chip with an integrated ARM-compatible central processing unit (CPU).

- The first-generation Raspberry Pi Model B was released in February 2012, followed by the simpler and cheaper Model A.
- In 2014, the Foundation released a board with an improved design, Raspberry Pi Model B+. These first-generation boards feature ARM11 processors, are approximately credit-card sized, and represent the standard mainline form factor. Improved A+ and B models were released within a year. A "Compute Module" was released in April 2014 for embedded applications.
- The Raspberry Pi 2 was released in February 2015 and initially featured a 900 MHz 32-bit quad-core ARM Cortex-A7 processor with 1 GB RAM. Revision 1.2 featured a 900 MHz 64-bit quad-core ARM Cortex-A53 processor (the same as that in the Raspberry Pi 3 Model B, but underclocked to 900 MHz).[25]
- The Raspberry Pi 3 Model B was released in February 2016 with a 1.2 GHz 64-bit quad core ARM Cortex-A53 processor, on-board 802.11n Wi-Fi, Bluetooth and USB boot capabilities.[26
- On Pi Day 2018, the Raspberry Pi 3 Model B+ was launched with a faster 1.4 GHz processor, a three-times faster Gigabit Ethernet (throughput limited to ca. 300 Mbit/s by the internal USB 2.0 connection), and 2.4 / 5 GHz dual-band 802.11ac Wi-Fi (100 Mbit/s).[27] Other features are Power over Ethernet (PoE) (with the add-on PoE HAT), USB boot and network boot (an SD card is no longer required).
- The Raspberry Pi 4 Model B was released in June 2019[1] with a 1.5 GHz 64-bit quad core ARM Cortex-A72 processor, on-board 802.11ac Wi-Fi, Bluetooth 5, full gigabit Ethernet (throughput not limited), two USB 2.0 ports, two USB 3.0 ports, 1–8 GB of RAM, and dual-monitor support via a pair of micro HDMI (HDMI Type D) ports for

up to 4K resolution. The version with 1 GB RAM has been abandoned and the prices of the 2 GB version have been reduced. The 8 GB version has a revised circuit board. The Pi 4 is also powered via a USB-C port, enabling additional power to be provided to downstream peripherals, when used with an appropriate PSU. But the Pi can only be operated with 5 volts and not 9 or 12 volts like other mini computers of this class. The initial Raspberry Pi 4 board has a design flaw where third-party e-marked USB cables, such as those used on Apple MacBooks, incorrectly identify it and refuse to provide power.[28][29] Tom's Hardware tested 14 different cables and found that 11 of them turned on and powered the Pi without issue.[30] The design flaw was fixed in revision 1.2 of the board, released in late 2019.[31] In mid-2021, Pi 4 B models appeared with the improved Broadcom BCM2711C0. The manufacturer is now using this chip for the Pi 4 B and Pi 400. However, the clock frequency of the Pi 4 B was not increased in the factory.

- The Raspberry Pi 400 was released in November 2020. A modern example of a keyboard computer, it features 4 GB of LPDDR4 RAM on a custom board derived from the existing Raspberry Pi 4 combined with a keyboard in a single case. The case was derived from that of the Raspberry Pi Keyboard.[32] A robust cooling solution (i.e. a broad metal plate) and an upgraded switched-mode power supply[33] allow the Raspberry Pi 400's Broadcom BCM2711C0 processor to be clocked at 1.8 GHz, which is 20% faster than the Raspberry Pi 4 upon which it is based.

Family	Model	SoC	Memory	Form Factor	Ethernet	Wireless	GPIO
Raspberry Pi	B	BCM2835	256 MB	Standard[a]	Yes	No	26-pin
			512 MB				
	A		256 MB		No		
	B+		512 MB		Yes		40-pin
	A+			Compact[b]	No		
Raspberry Pi 2	B	BCM2836 / 7	1 GB	Standard[a]	Yes	No	
Raspberry Pi Zero	Zero	BCM2835	512 MB	Ultra-compact[c]	No	No	
	W / WH					Yes	
	2 W	BCM2710A1[d][46]					
Raspberry Pi 3	B	BCM2837A0 / B0	1 GB	Standard[a]	Yes	Yes	
	A+	BCM2837B0	512 MB	Compact[b]	No	Yes[e]	
	B+		1 GB	Standard[a]	Yes[f]		
Raspberry Pi 4	B	BCM2711	1 GB	Standard[a]	Yes[g]	Yes[e]	
			2 GB				
			4 GB				
			8 GB				
	400		4 GB	Keyboard			
Raspberry Pi Pico	Pico	RP2040	264 KB	Pico[h]	No	No	
	W					Yes[i]	

Figure 4.1.2 Tabular form.

- Pi 1 Model B (2012)
- Pi 1 Model A (2013)
- Pi 1 Model B+ (2014)
- Pi 1 Model A+ (2014)
- Pi 2 Model B (2015)
- Pi Zero (2015)
- Pi 3 Model B (2016)
- Pi Zero W (2017)
- Pi 3 Model B+ (2018)
- Pi 3 Model A+ (2019)
- Pi 4 Model A (2019)
- Pi 4 Model B (2020)
- Pi 400 (2021)

Hardware

The Raspberry Pi hardware has evolved through several versions that feature variations in the type of the central processing unit, amount of memory capacity, networking support, and peripheral-device support.

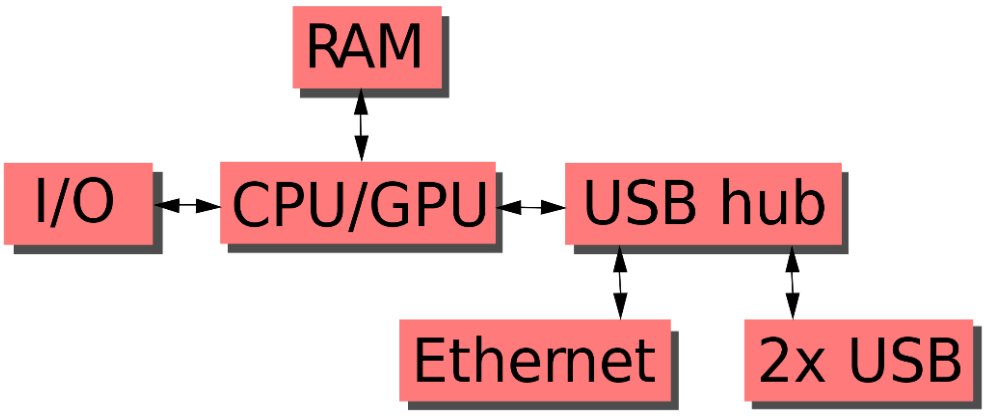


Figure 4.1.3 Connections.

This block diagram describes models B, B+, A and A+. The Pi Zero models are similar, but lack the Ethernet and USB hub components. The Ethernet adapter is internally connected to an additional USB port. In Model A, A+, and the Pi Zero, the USB port is connected directly to

the system on a chip (SoC). On the Pi 1 Model B+ and later models the USB/Ethernet chip contains a five-port USB hub, of which four ports are available, while the Pi 1 Model B only provides two. On the Pi Zero, the USB port is also connected directly to the SoC, but it uses a micro USB (OTG) port. Unlike all other Pi models, the 40 pin GPIO connector is omitted on the Pi Zero, with solderable through-holes only in the pin locations. The Pi Zero WH remedies this.

Processor speed ranges from 700 MHz to 1.4 GHz for the Pi 3 Model B+ or 1.5 GHz for the Pi 4; on-board memory ranges from 256 MB to 8 GB random-access memory (RAM), with only the Raspberry Pi 4 having more than 1 GB. Secure Digital (SD) cards in MicroSDHC form factor (SDHC on early models) are used to store the operating system and program memory, however some models also come with onboard eMMC storage[51] and the Raspberry Pi 4 can also make use of USB-attached SSD storage for its operating system.[52] The boards have one to five USB ports. For video output, HDMI and composite video are supported, with a standard 3.5 mm tip-ring-sleeve jack carrying mono audio together with composite video. Lower-level output is provided by a number of GPIO pins, which support common protocols like I²C. The B-models have an 8P8C Ethernet port and the Pi 3, Pi 4 and Pi Zero W have on-board Wi-Fi 802.11n and Bluetooth.

Processor

The Broadcom BCM2835 SoC used in the first generation Raspberry Pi[54] includes a 700 MHz 32-bit ARM1176JZF-S processor, VideoCore IV graphics processing unit (GPU),[55] and RAM. It has a level 1 (L1) cache of 16 KB and a level 2 (L2) cache of 128 KB. The level 2 cache is used primarily by the GPU. The SoC is stacked underneath the RAM chip, so only its edge is visible. The ARM1176JZ(F)-S is the same CPU used in the original iPhone,[56] although at a higher clock rate, and mated with a much faster GPU.

The earlier V1.1 model of the Raspberry Pi 2 used a Broadcom BCM2836 SoC with a 900 MHz 32-bit, quad-core ARM Cortex-A7 processor, with 256 KB shared L2 cache.[57] The Raspberry Pi 2 V1.2 was upgraded to a Broadcom BCM2837 SoC with a 1.2 GHz 64-bit quad-core ARM Cortex-A53 processor,[25] the same one which is used on the Raspberry Pi 3, but underclocked (by default) to the same 900 MHz CPU clock speed as the V1.1. The BCM2836 SoC is no longer in production as of late 2016.

The Raspberry Pi 3 Model B uses a Broadcom BCM2837 SoC with a 1.2 GHz 64-bit quad-core ARM Cortex-A53 processor, with 512 KB shared L2 cache. The Model A+ and B+ are 1.4 GHz[58][59][60]

The Raspberry Pi 4 uses a Broadcom BCM2711 SoC with a 1.5 GHz (later models: 1.8 GHz) 64-bit quad-core ARM Cortex-A72 processor, with 1 MB shared L2 cache.[61][62] Unlike previous models, which all used a custom interrupt controller poorly suited for virtualisation, the interrupt controller on this SoC is compatible with the ARM Generic Interrupt Controller (GIC) architecture 2.0, providing hardware support for interrupt distribution when using ARM virtualisation capabilities

The Raspberry Pi Zero and Zero W use the same Broadcom BCM2835 SoC as the first generation Raspberry Pi, although now running at 1 GHz CPU clock speed.

The Raspberry Pi Zero W 2 uses the RP3A0-AU CPU, a 1 GHz 64 bit ARM Cortex A53, on 512MB of SDRAM. Documentation states this "system-on-package" is a Broadcom BCM2710A1 package, using a BCM2837 Broadcom chip as core, which is an ARM v8 quad-core. The Raspberry Pi 3 also uses the BCM2837, but clocked at 1.2 GHz.

The Raspberry Pi Pico uses the RP2040 running at 133 MHz.

Performance

While operating at 700 MHz by default, the first generation Raspberry Pi provided a real-world performance roughly equivalent to 0.041 GFLOPS.[66][67] On the CPU level the performance is similar to a 300 MHz Pentium II of 1997–99. The GPU provides 1 Gpixel/s or 1.5 Gtexel/s of graphics processing or 24 GFLOPS of general purpose computing performance. The graphical capabilities of the Raspberry Pi are roughly equivalent to the performance of the Xbox of 2001.

Raspberry Pi 2 V1.1 included a quad-core Cortex-A7 CPU running at 900 MHz and 1 GB RAM. It was described as 4–6 times more powerful than its predecessor. The GPU was identical to the original.[57] In parallelised benchmarks, the Raspberry Pi 2 V1.1 could be up to 14 times faster than a Raspberry Pi 1 Model B+.

The Raspberry Pi 3, with a quad-core Cortex-A53 processor, is described as having ten times the performance of a Raspberry Pi 1.[69] Benchmarks showed the Raspberry Pi 3 to be approximately 80% faster than the Raspberry Pi 2 in parallelised tasks.[70]

The Raspberry Pi 4, with a quad-core Cortex-A72 processor, is described as having three times the performance of a Raspberry Pi 3.[1]

Networking

The Model A, A+ and Pi Zero have no Ethernet circuitry and are commonly connected to a network using an external user-supplied USB Ethernet or Wi-Fi adapter. On the Model B and B+ the Ethernet port is provided by a built-in USB Ethernet adapter using the SMSC LAN9514 chip.[83] The Raspberry Pi 3 and Pi Zero W (wireless) are equipped with 2.4 GHz WiFi 802.11n (150 Mbit/s) and Bluetooth 4.1 (24 Mbit/s) based on the Broadcom BCM43438 FullMAC chip with no official support for monitor mode (though it was implemented through unofficial firmware patching[84]) and the Pi 3 also has a 10/100 Mbit/s Ethernet port. The Raspberry Pi 3B+ features dual-band IEEE 802.11b/g/n/ac WiFi, Bluetooth 4.2, and Gigabit Ethernet (limited to approximately 300 Mbit/s by the USB 2.0 bus between it and the SoC). The Raspberry Pi 4 has full gigabit Ethernet (throughput is not limited as it is not funnelled via the USB chip.)

4.1.2 : Camera Module

Figure 4.1.4 Raspberry pi 3B+ Camera Module.

The Raspberry Pi camera module can be used to take high-definition video, as well as stills photographs. It's easy to use for beginners, but has plenty to offer advanced users if you're looking to expand your knowledge. There are lots of examples online of people using it for

time-lapse, slow-motion and other video cleverness. You can also use the libraries we bundle with the camera to create effects.

If you're interested in the nitty-gritty, you'll want to know that the module has a five megapixel fixed-focus camera that supports 1080p30, 720p60 and VGA90 video modes, as well as stills capture. It attaches via a 15cm ribbon cable to the CSI port on the Raspberry Pi. It can be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the Picamera Python library.

The camera module is very popular in home security applications, and in wildlife camera traps.

You can also use it to take snapshots.

Features

- 5MP sensor
- Wider image, capable of 2592x1944 stills, 1080p30 video
- 1080p video supported
- CSI
- Size: 25 x 20 x 9 mm

Camera Details

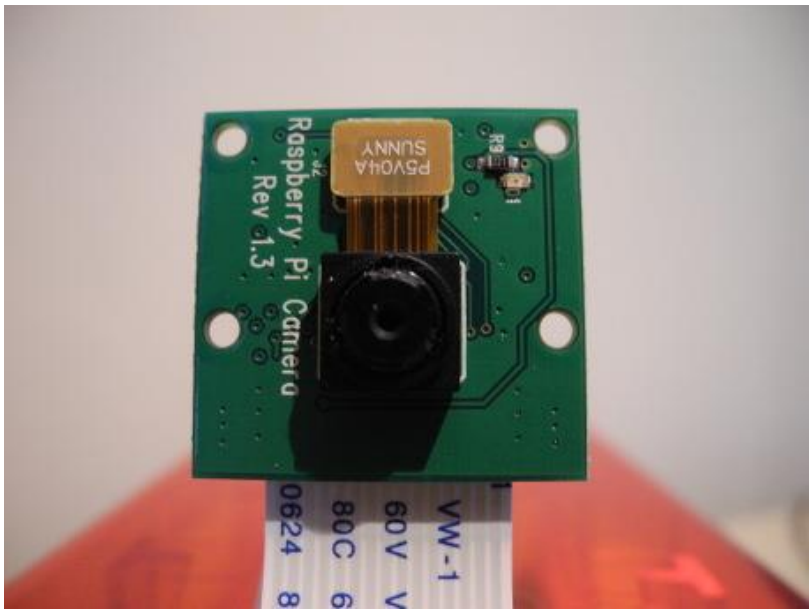
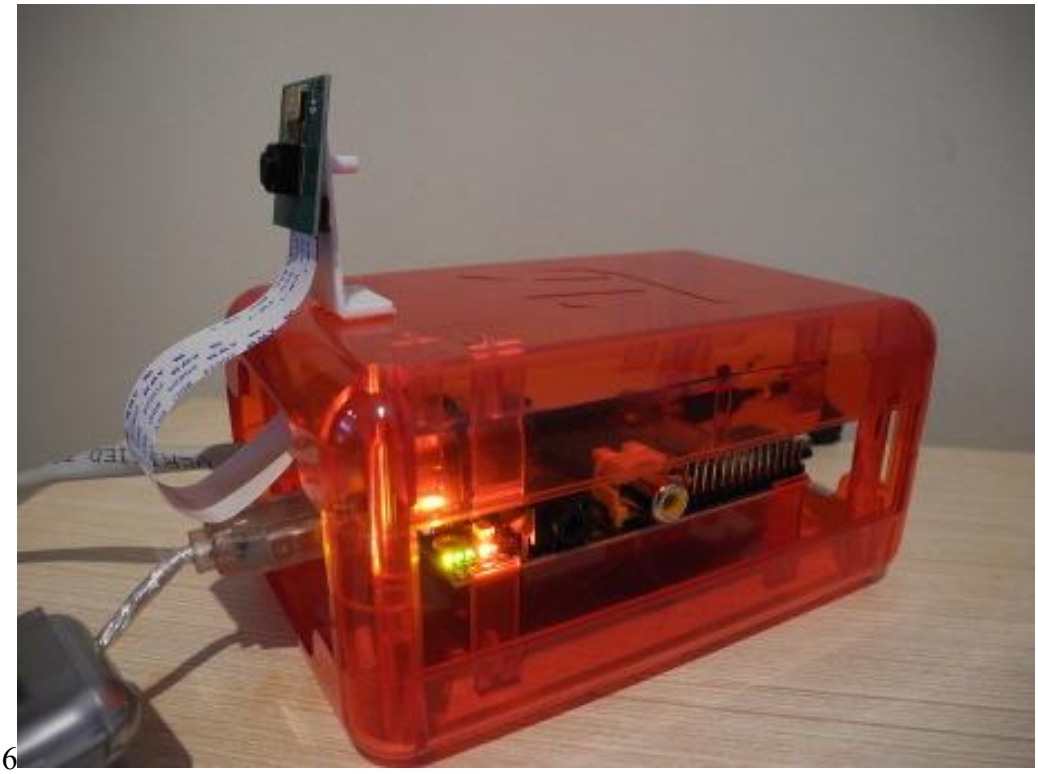


Figure 4.1.5. Camera details.

The camera consists of a small (25mm by 20mm by 9mm) circuit board, which connects to the Raspberry Pi's Camera Serial Interface (CSI) bus connector via a flexible ribbon cable. The camera's image sensor has a native resolution of five megapixels and has a fixed focus lens. The software for the camera supports full resolution still images up to 2592x1944 and video resolutions of 1080p30, 720p60 and 640x480p60/90. The camera module is shown above:

Installation involves connecting the ribbon cable to the CSI connector on the Raspberry Pi board. This can be a little tricky, but if you watch the videos that demonstrate how it is done, you shouldn't have any trouble.

When you purchase the camera, you will receive a small camera board and cable. You'll want to devise some method of supporting the camera in order to use it. Some camera stands and Raspberry Pi cases are now available. You can also rig up something simple yourself if you wish. I attached mine to a case using a small piece of plastic and double-sided tape, as shown below:



6

Figure 4.1.6 Connection kit.

Once the hardware is set up, you can move on to configuring the software.

Connect to the camera

- The flex cable inserts into the connector situated between the Ethernet and HDMI ports, with the silver connectors facing the HDMI port. The flex cable connector should be opened by pulling the tabs on the top of the connector upwards then towards the Ethernet port. The flex cable should be inserted firmly into the connector, with care taken not to bend the flex at too acute an angle. The top part of the connector should then be pushed towards the HDMI connector and down, while the flex cable is held in place.
- Update the SD card

In order to use the camera you must be using a recent operating system that knows that the camera exists. The easiest way to do this is to grab the latest Raspbian image from the RaspberryPi.org site and create a fresh SD card.

- Enable camera in raspi-config settings

Reboot. If you are using a fresh image the raspi-config utility should load. If it doesn't then you can run it manually using : sudo raspi-config Select the "Camera" option and press "Enter".

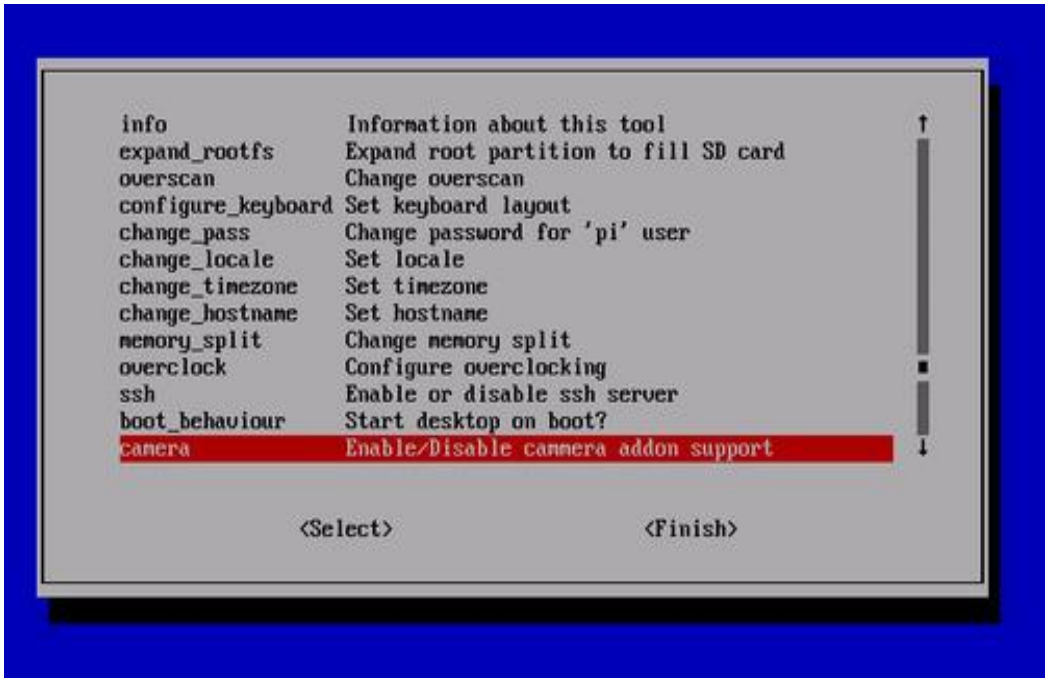


Figure 4.1.7 Select the "Camera" option and press "Enter".

Select “Enable” and press “Enter”.

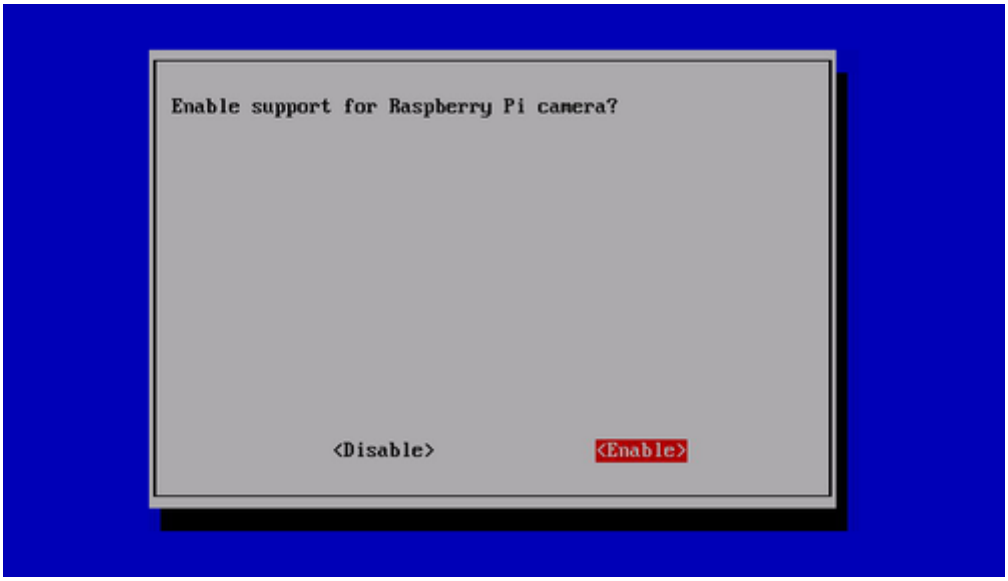


Figure 4.1.8 Select “Enable” and press “Enter”.

Select “Yes” and press “Enter”. Your Pi will reboot.

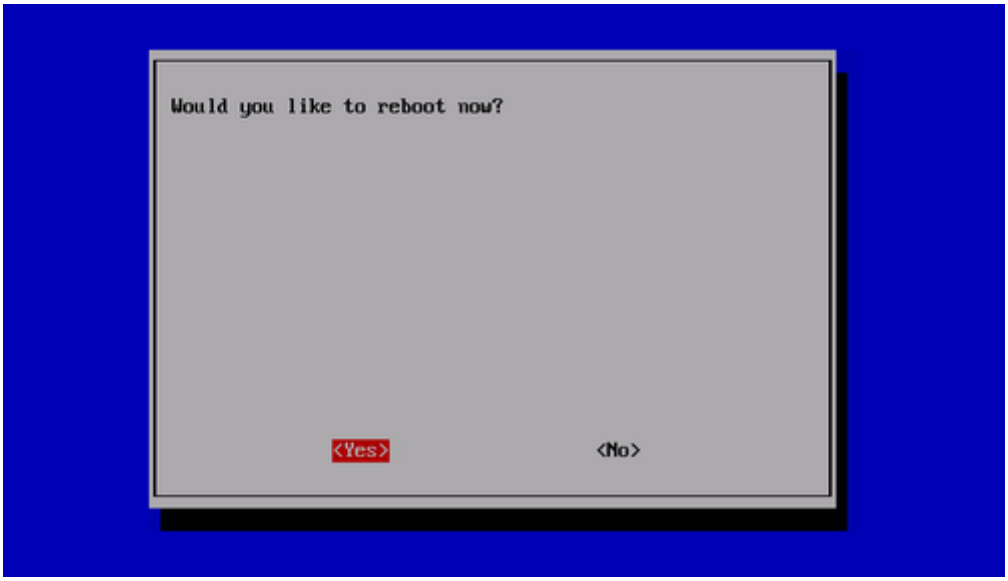


Figure 4.1.9 Select “Yes” and press “Enter”. Your Pi will reboot.

Updating your operating and enabling the camera using raspi-config did two things. It told your Pi that there is a camera attached and it added two command line utilities. raspistill raspivid These allow you to capture still photos and HD video respectively.

4.2 SOFTWARE SPECIFICATIONS

4.2.1 GOOGLE FIRE BASE

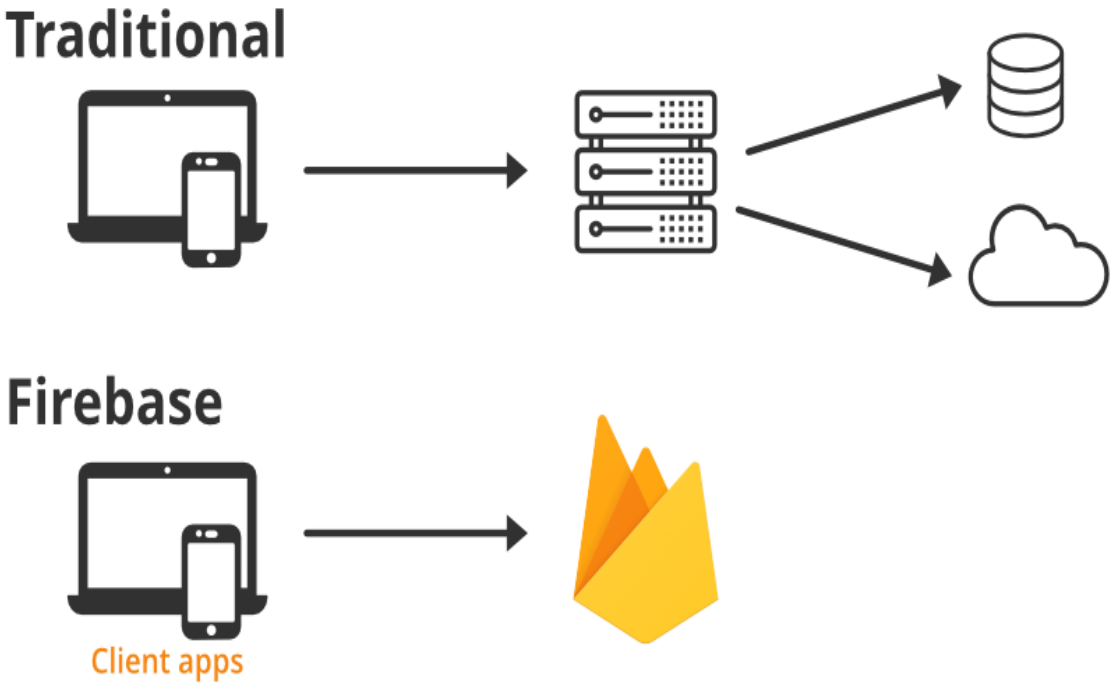


Figure 4.2.1 Google Fire Base.

Firebase is a set of backend cloud computing services and application development platforms provided by Google. It hosts databases, services, authentication, and integration for a variety of applications, including Android, iOS, JavaScript, Node.js, Java, Unity, PHP, and C++.

Firebase is a product of Google which helps developers to build, manage, and grow their apps easily. It helps developers to build their apps faster and in a more secure way. No programming is required on the firebase side which makes it easy to use its features more efficiently. It provides services to android, ios, web, and unity. It provides cloud storage. It uses NoSQL for the database for the storage of data.

Brief History of Firebase:

Firebase initially was an online chat service provider to various websites through API and ran with the name Envolve. It got popular as developers used it to exchange application data like a game state in real time across their users more than the chats. This resulted in the separation of the Envolve architecture and it's chat system. The Envolve architecture was further evolved by it's founders James Tamplin and Andrew Lee,to what modern day Firebase is in the year 2012.

Features of Firebase:

Mainly there are 3 categories in which firebase provides its services.

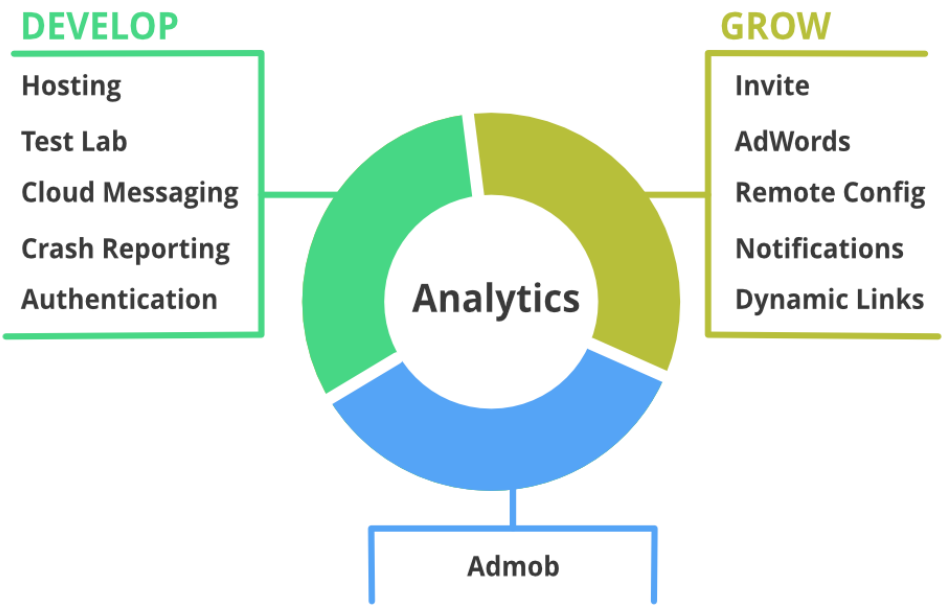


Figure 4.2.2 Categories in which firebase provides its services.

Build better applications

This feature mainly includes backend services that help developers to build and manage their applications in a better way. Services included under this feature are :

- **Realtime Database**: The Firebase Realtime Database is a cloud-based NoSQL database that manages your data at the blazing speed of milliseconds. In simplest term, it can be considered as a big JSON fil

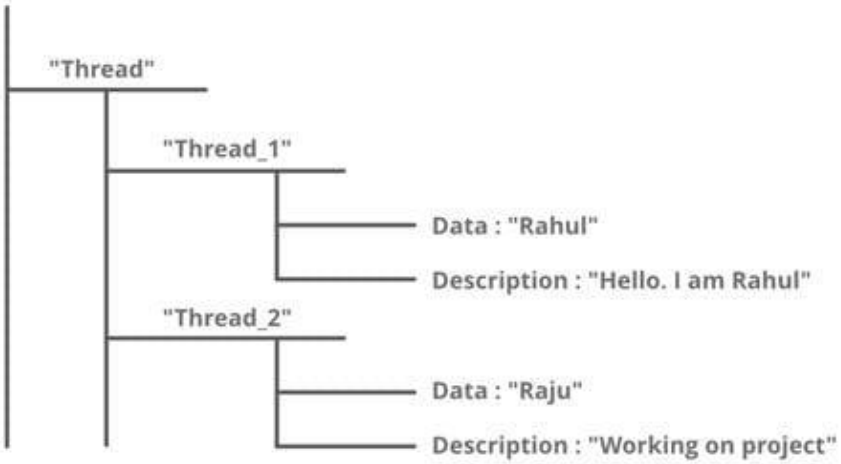


Figure 4.2.3 Realtime Database.

- **Cloud Firestore**: The cloud Firestore is a NoSQL document database that provides services like store, sync, and query through the application on a global scale. It stores data in the form of objects also known as Documents. It has a key-value pair and can store all kinds of data like, strings, binary data, and even JSON trees.

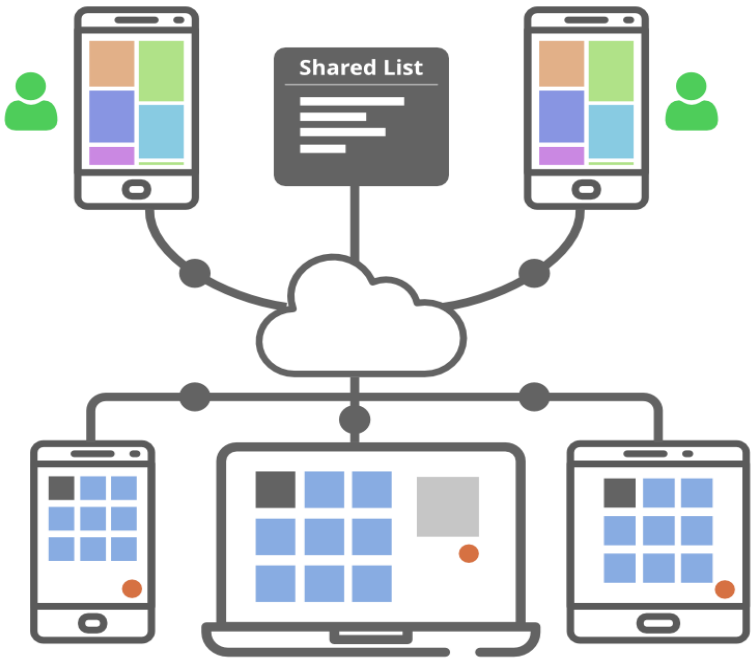


Figure 4.2.4 Cloud firestore.

- **Authentication**: Firebase Authentication service provides easy to use UI libraries and SDKs to authenticate users to your app. It reduces the manpower and effort required to develop and maintain the user authentication service. It even handles tasks like merging accounts, which if done manually can be hectic.

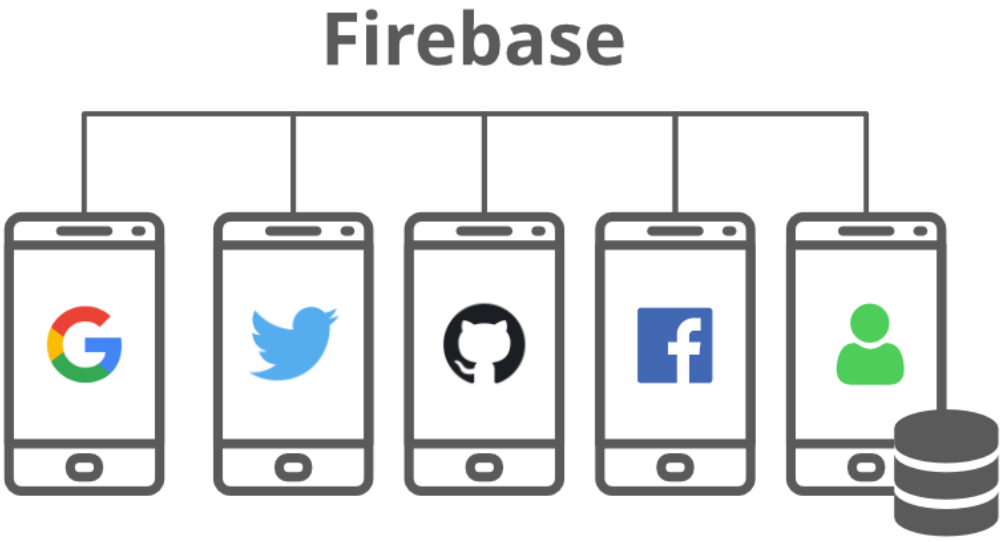


Figure 4.2.5 Authentication.

Remote Config: The remote configuration service helps in publishing updates to the user immediately. The changes can range from changing components of the UI to changing the behavior of the applications. These are often used while publishing seasonal offers and contents to the application that has a limited life.

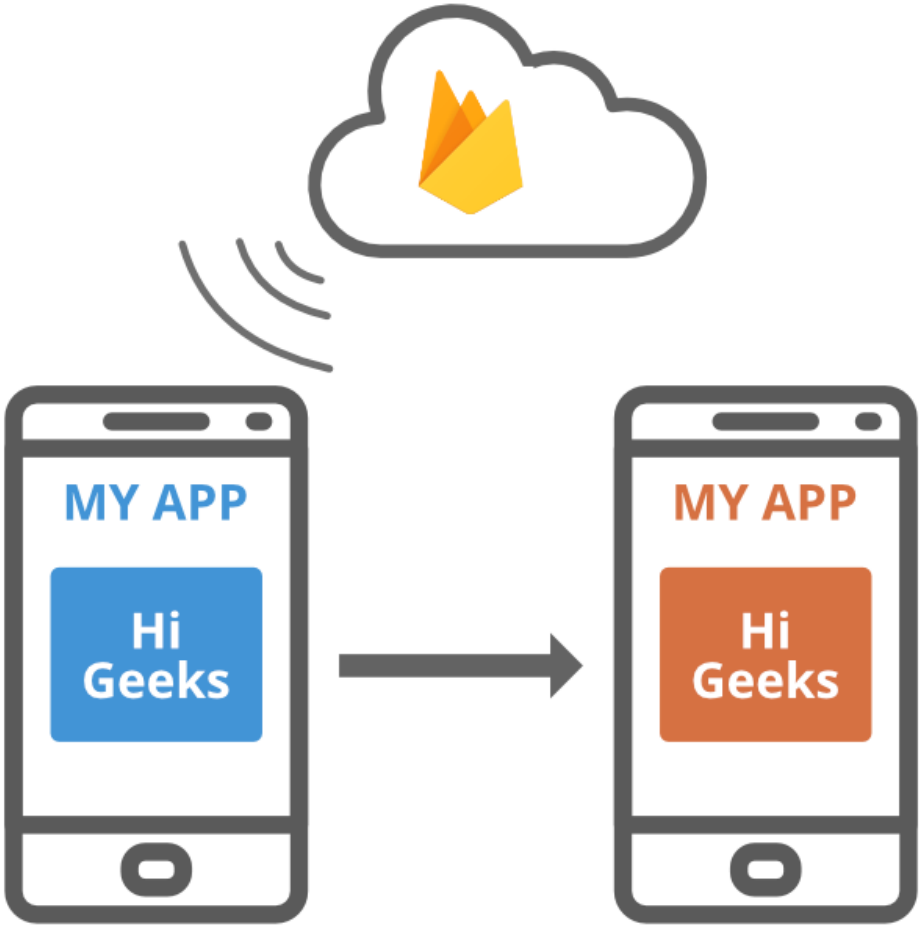


Figure 4.2.6 Remote configuration.

- **Hosting:** Firebase provides hosting of applications with speed and security. It can be used to host Stati or Dynamic websites and microservices. It has the capability of hosting an application with a single command.
- **Firebase Cloud Messaging(FCM):** The FCM service provides a connection between the server and the application end users, which can be used to receive and send messages and notifications. These connections are reliable and battery-efficient.

  The service is provided by firebase, a subsidiary of Google. On October 21, 2014, Firebase announced it had been acquired by Google for an undisclosed amount. The

official Google Cloud Messaging website points to Firebase Cloud Messaging (FCM) as the new version of GCM. Firebase is a mobile platform which supports users in developing mobile and web applications. Firebase Cloud Messaging is one of many products which are part of the Firebase platform. On the platform users can integrate and combine different Firebase features in both web and mobile applications.

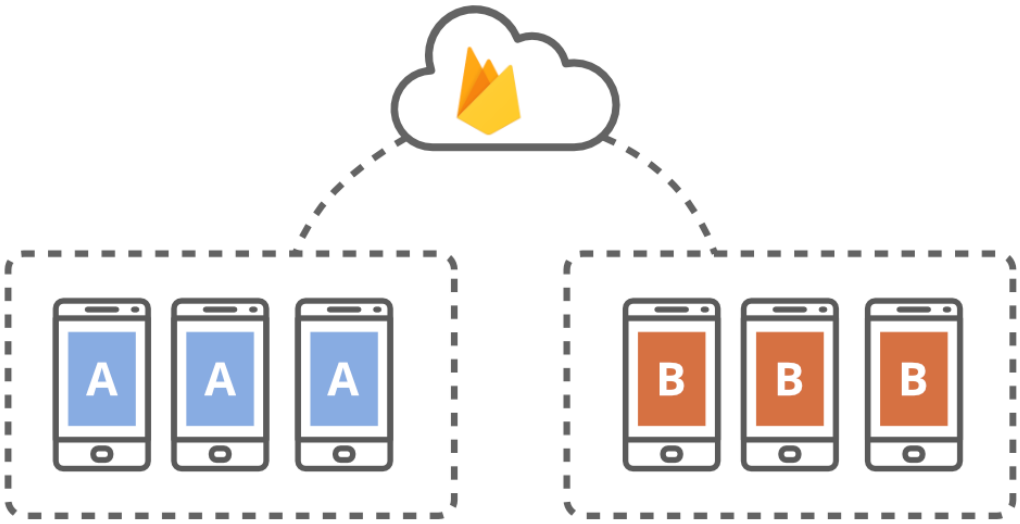


Figure 4.2.7 Firebase Cloud Messaging.

CHAPTER 5

RESULTS AND DISCUSSION

After the module processes the received inputs, the face of the person in front of the camera is recognized and the outputs are generated.

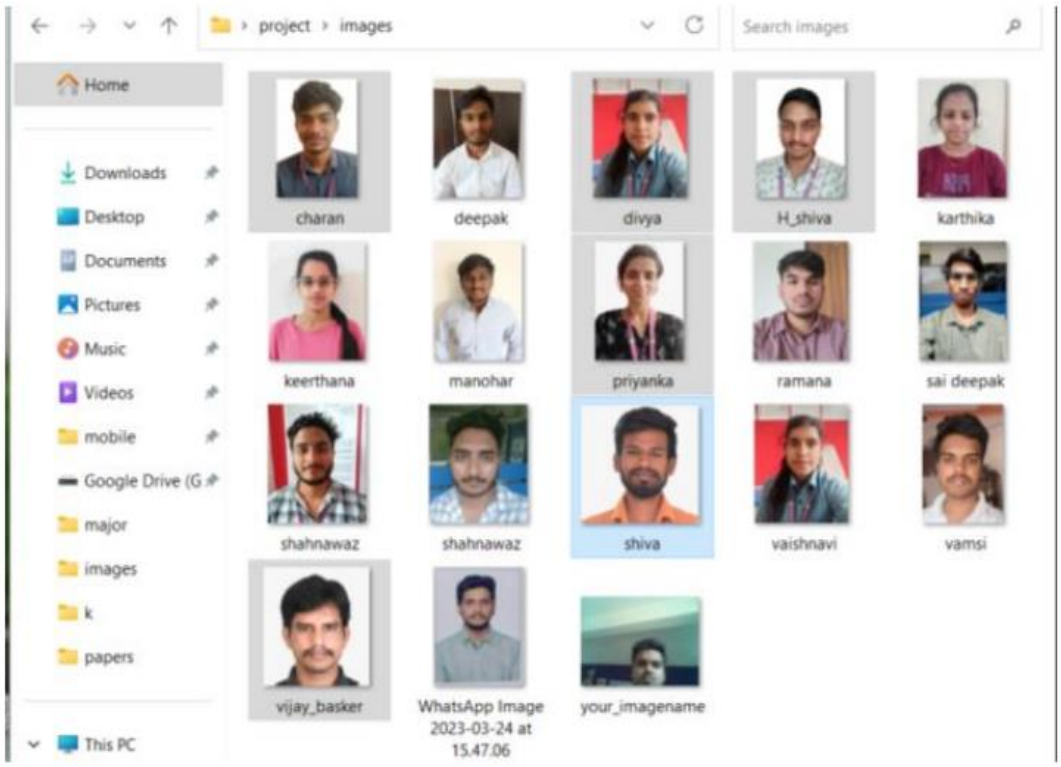


Figure 5.1 Collection of data of n number of students.

The data sets are stored in the local storage path and uploaded to firebase, which we are using as a database here. Here we have taken images of around 18 students and each image has the student's name to it.

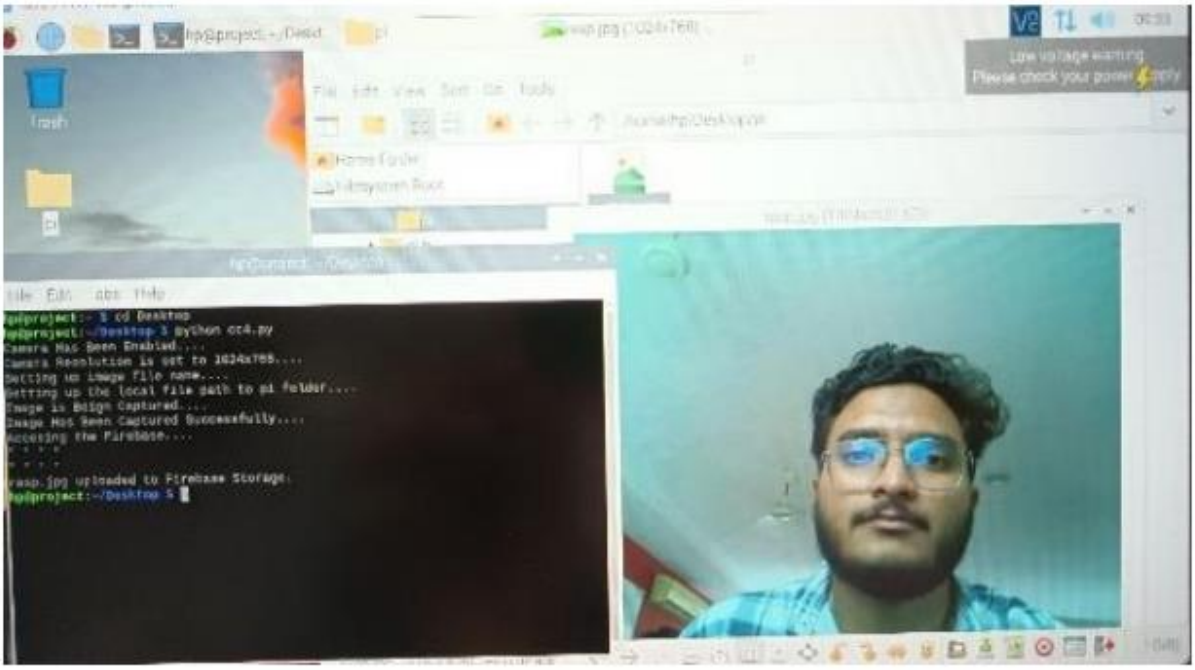


Figure 5.2 Captured image through pi 3b camera module.

The above figure shows us how the inputs are taken. We have used a raspberry pi 3b camera module, an external camera connected to raspberry pi to take the photo. It is mainly to capture a more clarity photo of the subject shown above. It has a default resolution of 1024*768 but can be changeable.

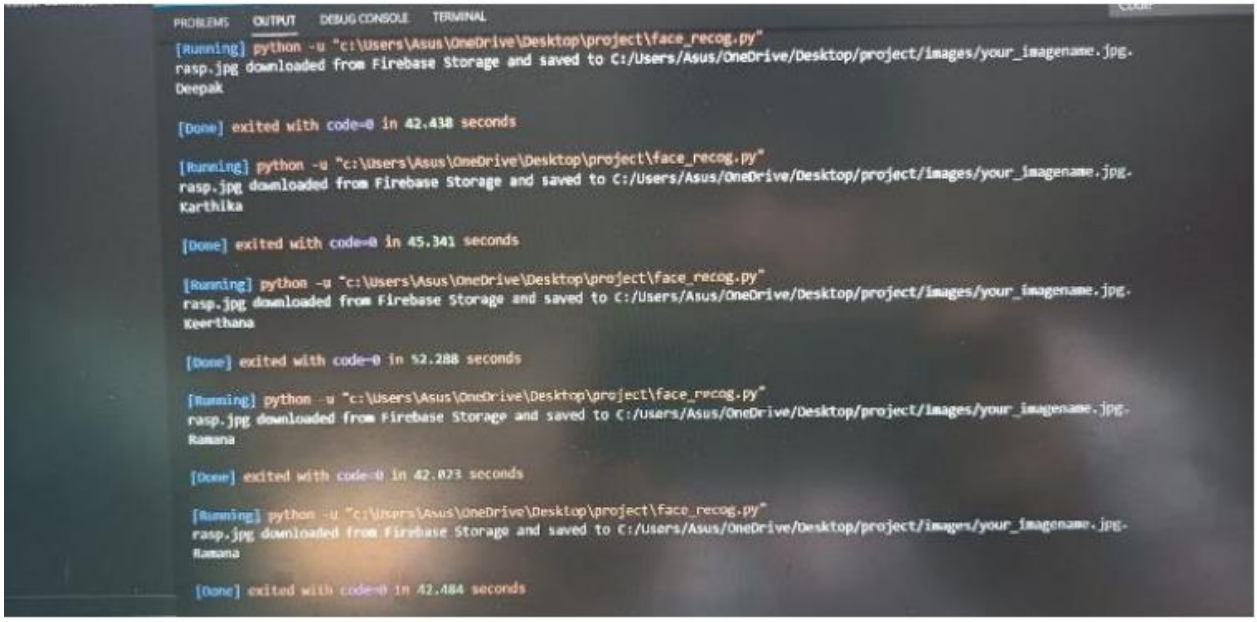


Figure 5.3 Processing time of multiple inputs.

As we can observe, we have processed the inputs multiple times to check for any errors or time delay. But we can see here that through a working prototype we were able to process the input image under a minute. The time could be reduced even further with the use of a better hardware.

Table 5.1 Output data of students that are marked present.

Name	Branch	Designation	Roll Number	Time	Date
Vaishnavi	ECE	Student	19qm1a0467	Fri 08:51:27	May 12 2023
Sai Deepak	ECE	Student	19qm1a0444	Fri 08:53:56	May 12 2023
Vamsi	ECE	Student	19qm1a0409	Fri 09:44:56	May 12 2023
Hemanth Rao	ECE	Student	19qm1a0410	Sat 12:03:01	May 13 2023
Deepak	ECE	Student	19qm1a0456	Sat 10:03:44	May 13 2023
Vijay	ECE	Staff	Kgr128	Mon 11:44:32	May 15 2023
Keerthana	ECE	Student	20qm5a0412	Tue 09:44:56	May 16 2023
Ramana	ECE	Student	19qm1a0446	Wed 14:55:14	May 17 2023

After validating the face, a file with name/roll number, date and time are generated and will be store in the desired location. We used cloud as the storage path allowing the data in the file to be updated continuously without any file redundancy.

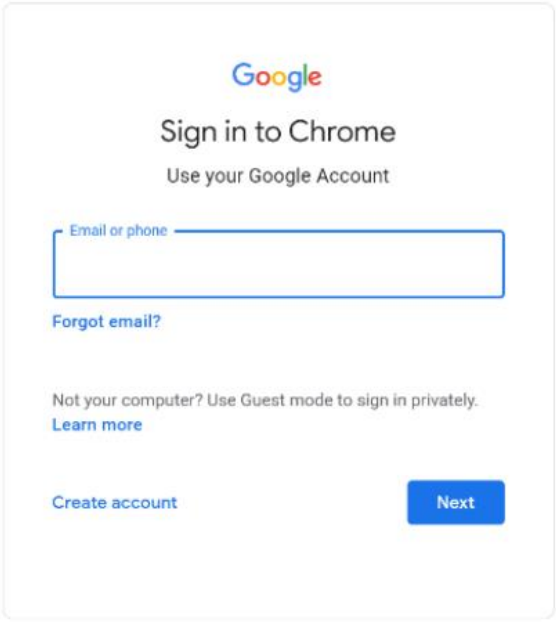


Figure 5.4 Login credentials

As the data is stored in cloud, it is accessible to anyone with right credentials. The students can check if their attendance is marked correctly and the faculty can make a group sheet on a monthly basis to check on each student's attendance percentage. It allows them to know the total classes held vs total classes attended.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

The work titled “Attendance Management System based on Face Recognition” is a model to manage student attendance using raspberry pi and firebase as its main base. It helps in recording day to day student attendance either present/absent by using live facial recognition in the respective classroom. Normally, it reduces the task of a teacher to take attendance by a roll call. It in turn helps in time management and reduces the chance to miss a student attendance. The system uses CNN algorithm, which is found to have high flexibity over facial recognition even within multiple crowds and with minimum data provided. The data on a every day basis is stored locally and in the cloud with an almost no particular time delay. It can accesed by the people with the right credentials to cross-check or further process the data based on our requirement. As it is soleley made for educational instituations by keeping in mind the management needs, it is safe, secure and also increases productivity. It can be further extended to be used in other commertial applications by changing a few things and keeping its core process constant. Overall, the proposed system works for the benefit of students, where attendance is an important part of their day to day evaluation.

6.2 FUTURE SCOPE

Further work can be done on this project to alert the student by sending SMS regarding the attendance. For this purpose GSM module can be used. SMS alert can be given to the parent of the student.

REFERENCES

[1] S. Kakarla, P. Gangula, M. S. Rahul, C. S. C. Singh and T. H. Sarma, "Smart Attendance Management System Based on Face Recognition Using CNN," 2020 IEEE-HYDCON, Hyderabad, India, 2020, pp. 1-5, doi: 10.1109/HYDCON48903.2020.9242847.

[2] https://arxiv.org/abs/1511.08458

[3] H. -C. Shin et al., "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," in IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1285-1298, May 2016, doi: 10.1109/TMI.2016.2528162.

[4] B. Tej Chinimilli, A. T., A. Kotturi, V. Reddy Kaipu and J. Varma Mandapati, "Face Recognition based Attendance System using Haar Cascade and Local Binary Pattern Histogram Algorithm," 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 2020, pp. 701-704, doi: 10.1109/ICOEI48184.2020.9143046.

[5] S. Wu and H. Nagahashi, "Parameterized AdaBoost: Introducing a Parameter to Speed Up the Training of Real AdaBoost," in IEEE Signal Processing Letters, vol. 21, no. 6, pp. 687-691, June 2014, doi: 10.1109/LSP.2014.2313570.

[6] L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," *2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*, Yangzhou, China, 2017, pp. 483-487, doi: 10.1109/ICEMI.2017.8265863.

[7] M. S. Mubarak Alburaiki, G. Md Johar, R. A. Abbas Helmi and M. Hazim Alkawaz, "Mobile Based Attendance System: Face Recognition and Location Detection using Machine Learning," 2021 IEEE 12th Control and System Graduate Research Colloquium (ICSGRC), 2021, pp. 177- 182, doi: 10.1109/ICSGRC53186.2021.9515221.

[8] Q. Y. Tan, P. S. Joseph Ng and K. Y. Phan, "JomRFID Attendance Management System," 2021 Innovations in Power and Advanced Computing Technologies (i-PACT), 2021, pp. 1-6, doi: 10.1109/i-PACT52855.2021.9696816.

[9] M. R, M. D and R. P, "Classroom Attendance Monitoring Using CCTV," 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), 2020, pp. 1-4, doi: 10.1109/ICSCAN49426.2020.9262436.

[10] R. P. Vandana, P. S. Venugopala and B. Ashwini, "Neural Network based Biometric Attendance System," 2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), 2021, pp. 84- 87, doi: 10.1109/DISCOVER52564.2021.9663661.

[11] P. Sarath Krishnan and A. Manikuttan, "Attendance Management System Using Facial Recognition," 2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS), 2022, pp. 1-6, doi: 10.1109/IC3SIS54991.2022.9885693.

[12] G. Sittampalam and N. Ratnarajah, "SAMS: An IoT Solution for Attendance Management in Universities," TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), 2019, pp. 251-256, doi: 10.1109/TENCON.2019.8929616.

[13] A. Hake, A. Samanta, P. Kasambe and R. Sutar, "Automatic Attendance Marker Using Beacon technology," 2022 IEEE Region 10 Symposium (TENSYMP), 2022, pp. 1-5, doi: 10.1109/TENSYMP54529.2022.9864555.

[14] A. Sukmandhani and I. Sutedja, "Face Recognition Method for Online Exams," 2019 International Conference on Information Management and Technology (ICIMTech), 2019, pp. 175-179, doi: 10.1109/ICIMTech.2019.8843831.

[15] https://www.geeksforgeeks.org/introduction-convolutionneural-network/https://www.geeksforgeeks.org/introductionconvolution-neural-network/

[16] https://www.codemag.com/Article/2205081/ImplementingFace-Recognition-Using-Deep-Learning-and-Support-VectorMachines#:~:text=Deep%20Learning%20%2D%20Convolutional%20Neural%20Network,used%20in%20face%20recognition%20 software.

[17] https://miro.medium.com/v2/resize:fit:1400/1*fQBZTdPk_Yz aR7If7Sjzxg.png

[18] https://medium.com/analytics-vidhya/haar-cascadesexplained-38210e57970d.

[19] M. S. Mohd Azmi, M. H. Mohamed Zabil, K. C. Lim, R. F. Raja Azman, N. A. Noor Adnan and M. A. S. Mohamed Azman, "UNITEN Smart Attendance System (UniSas) Using Beacons Sensor," 2018 IEEE Conference on e-Learning, e-Management and e-Services (IC3e), Langkawi, Malaysia, 2018, pp. 35-39, doi: 10.1109/IC3e.2018.8632631.

[20] T. Adiono, D. Setiawan, Maurizfa, J. William and N. Sutisna, "Cloud Based User Interface Design for Smart Student Attendance System," 2021 International

Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2021, pp. 1-5, doi: 10.1109/ISESD53023.2021.9501878.

[21] K. Navin, A. Shanthini and M. B. M. Krishnan, "A mobile based smart attendance system framework for tracking field personals using a novel QR code based technique," 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), Bengaluru, India, 2017, pp. 1540-1543, doi: 10.1109/SmartTechCon.2017.8358623.

[22] A. Bejo, R. Winata and S. S. Kusumawardani, "Prototyping of Class-Attendance System Using Mifare 1K Smart Card and Raspberry Pi 3," 2018 International Symposium on Electronics and Smart Devices (ISESD), Bandung, Indonesia, 2018, pp. 1-5, doi: 10.1109/ISESD.2018.8605442.

[23] Z. Xu, P. Chen, W. Zhang, X. Liu and H. Wu, "Research on Mobile Phone Attendance Positioning System Based on Campus Network," 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), Xiangtan, China, 2019, pp. 387-389, doi: 10.1109/ICSGEA.2019.00094.

[24] R. V. Imbar, B. Renaldy Sutedja and M. Christianti, "Smart Attendance Recording System using RFID and e-Certificate using QR Code-based Digital Signature," 2021 International Conference on ICT for Smart Society (ICISS), Bandung, Indonesia, 2021, pp. 1-5, doi: 10.1109/ICISS53185.2021.9533199.

[25] R. Vijayakumar, M. Poornima, S. Divyapriya and T. Selvaganapathi, "Automated Student Attendance Tracker for End Semester Examination using Face Recognition System," 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2022, pp. 1566-1570, doi: 10.1109/ICOSEC54921.2022.9952035.

[26] S. Anand, K. Bijlani, S. Suresh and P. Praphul, "Attendance Monitoring in Classroom Using Smartphone & Wi-Fi Fingerprinting," 2016 IEEE Eighth International Conference on Technology for Education (T4E), Mumbai, India, 2016, pp. 62-67, doi: 10.1109/T4E.2016.021.

[27] S. K. Baharin, Z. Zulkifli and S. B. Ahmad, "Student Absenteeism Monitoring System Using Bluetooth Smart Location-Based Technique," 2020 International Conference on Computational Intelligence (ICCI), Bandar Seri Iskandar, Malaysia, 2020, pp. 109-114, doi: 10.1109/ICCI51257.2020.9247809.

[28] A. Shene, J. Aldridge and H. Alamleh, "Privacy-Preserving Zero-effort Class Attendance Tracking System," 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Toronto, ON, Canada, 2021, pp. 1-4, doi: 10.1109/IEMTRONICS52119.2021.9422481.

[29] M. Hu and H. Li, "Application of Location-Controlled Mobile Attendance Recording System in College Classroom Teaching," 2020 International Symposium on Educational Technology (ISET), Bangkok, Thailand, 2020, pp. 18-22, doi: 10.1109/ISET49818.2020.00014.

[30] S. Sharma, S. Monika, S. V. S. Prasad, K. Dasari and S. Kamaganikuntla, "RFID Based Low Cost Attendance Recording and Proxy Avoidance System," 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC), Bhubaneswar, India, 2022, pp. 1-5, doi: 10.1109/ASSIC55218.2022.10088295.

[31] V. Mishra, S. Raj, T. Singhal and C. Sankhla, "Intelligent Face Recognition based attendance system," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-4, doi: 10.1109/ICRITO56286.2022.9964862.

[32] P. Gupta and B. Singh, "A New Way of Recording Attendance of the Students using Face Recognition System," 2022 5th International Conference on Contemporary Computing and Informatics (IC3I), Uttar Pradesh, India, 2022, pp. 578-583, doi: 10.1109/IC3I56241.2022.10073382.

[33] D. Yadav, S. Maniar, K. Sukhani and K. Devadkar, "In-Browser Attendance System using Face Recognition and Serverless Edge Computing," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2021, pp. 01-06, doi: 10.1109/ICCCNT51525.2021.9580042.

[34] Efanntyo and A. R. Mitra, "Masked Face Recognition by Applying SSD and ResNet Model for Attendance System," 2021 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA), Surabaya, Indonesia, 2021, pp. 234-238, doi: 10.1109/ICAMIMIA54022.2021.9807814.

SOURCE CODE

RASPBERRY PI TO DATA FIREBASE

```
import firebase_admin

from firebase_admin import credentials, storage

from google.cloud import storage as gcs

from picamera import PiCamera

from time import sleep

# Initialize the PiCamera module

camera = PiCamera()

print("Camera Has Been Enabled....")

# Set the camera resolution

camera.resolution = (1024, 768)

print("Camera Resolution is set to 1024x768....")

# Set the file name of the image to be captured

image_file_name = 'rasp.jpg'

print("Setting up image file name....")

# Set the local file path of the image file you want to upload

local_file_path = f'/home/hp/Desktop/pi/{image_file_name}'

print("Setting up the local file path to pi folder....")

# Wait for the camera to warm up

print("Image is Beign Captured....")

sleep(5)

# Capture the image from the camera

camera.capture(local_file_path)
```

```
print("Image Has Been Captured Successfully....")

# Load the Firebase service account credentials from a JSON file

cred = credentials.Certificate('/home/hp/Desktop/icps-9cc0a-firebase-adminsdk-8nlxz-c92932012f.json')

print("Accesing the Firebase....")

# Initialize the Firebase app with the service account credentials

firebase_admin.initialize_app(cred, {'storageBucket': 'icps-9cc0a.appspot.com'})

# Get a reference to the Firebase storage bucket

bucket = storage.bucket()

print("* * * *")

# Upload the image file to Firebase storage

blob = bucket.blob(image_file_name)

print("* * * *")

blob.upload_from_filename(local_file_path)

print(f'{image_file_name} uploaded to Firebase Storage.')
```

ACCESSING DATA FROM THE FIREBASE

```
import face_recognition

import cv2

import numpy as np

import datetime

import pandas as pd

import firebase_admin

from firebase_admin import credentials, storage
```

```
from google.cloud import storage as gcs
import requests
from datetime import datetime
from datetime import datetime, timedelta

cred = credentials.Certificate('C:/Users/Asus/OneDrive/Desktop/project/icps-9cc0a-firebase-adminsdk-8nlxz-c92932012f.json')

# Initialize the Firebase app with the service account credentials
firebase_admin.initialize_app(cred)
image_file_name = 'rasp.jpg'
bucket_name = 'icps-9cc0a.appspot.com'
# Get a reference to the Firebase storage bucket and image file
bucket = storage.bucket(bucket_name)
blob = bucket.blob(image_file_name)

# Set the local file path where you want to save the image
local_file_path = 'C:/Users/Asus/OneDrive/Desktop/project/images/your_imagename.jpg'

# Download the image file from Firebase storage
expiration_time = datetime.utcnow() + timedelta(minutes=10)
url = blob.generate_signed_url(expiration_time,method='GET')
response = requests.get(url)
with open(local_file_path, 'wb') as f:
    f.write(response.content)
```

```
print(f'{image_file_name} downloaded from Firebase Storage and saved to {local_file_path}.')


deepak_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/deepak.jpg")
deepak_encoding = face_recognition.face_encodings(deepak_face)[0]
karthika_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/karthika.jpg")
karthika_encoding = face_recognition.face_encodings(karthika_face)[0]
keerthana_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/keerthana.jpg")
keerthana_encoding = face_recognition.face_encodings(keerthana_face)[0]
manohar_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/manohar.jpg")
manohar_encoding = face_recognition.face_encodings(manohar_face)[0]
ramana_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/ramana.jpg")
ramana_encoding = face_recognition.face_encodings(ramana_face)[0]
sai_deepak_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/sai deepak.jpg")
sai_deepak_encoding = face_recognition.face_encodings(sai_deepak_face)[0]
```

```
shahnawaz_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/shahnawaz.jpg")

shahnawaz_encoding = face_recognition.face_encodings(shahnawaz_face)[0]

vaishnavi_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/vaishnavi.jpg")

vaishnavi_encoding = face_recognition.face_encodings(vaishnavi_face)[0]

vamsi_face = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/vamsi.jpg")

vamsi_encoding = face_recognition.face_encodings(vamsi_face)[0]

hr = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/WhatsApp Image 2023-03-24 at 15.47.06.jpeg")

hre = face_recognition.face_encodings(hr)[0]


known_face_encodings = [
    deepak_encoding,
    karthika_encoding,
    keerthana_encoding,
    manohar_encoding,
    ramana_encoding,
    sai_deepak_encoding,
    shahnawaz_encoding,
    vaishnavi_encoding,
    vamsi_encoding,
```

```
    hre
]
known_face_names = [
    "Deepak",
    "Karthika",
    "Keerthana",
    "Manohar",
    "Ramana",
    "Sai Deepak",
    "shahnawaz",
    "Vasihnavi",
    "Vamsi",
    "Hemanth Rao"
]
roll_no = [
    "19qm1a0447",
    "19qm1a0456",
    "xxxxxxxx",
    "xxxxxxxx",
    "19qm1a0446",
    "19qm1a0444",
    "xxxxxxxx",
    "19qm1a0467",
    "xxxxxxxx",
```

```
    "xxxxxxxx"
]
designation = [
    "Student",
    "Student",
    "Faculty",
    "Faculty",
    "Student",
    "Student",
    "Faculty",
    "Student",
    "Faculty",
    "Faculty"
]
Branch = [
    "ECE"
]
m2 = face_recognition.load_image_file("C:/Users/Asus/OneDrive/Desktop/project/images/your_imagename.jpg")
m2_encoding = face_recognition.face_encodings(m2)[0]
face_distances = face_recognition.face_distance(known_face_encodings, m2_encoding)
name = known_face_names[np.argmin(face_distances)]
index = np.argmin(face_distances)
print("Name: {}".format(name))
```

```
print("Branch: {}".format(Branch[0]))

print("Roll Number: {}".format(roll_no[index]))

print(" Designation: {}".format(designation[index]))

x = datetime.now()

data = {

    'Name' : [name],

    'Branch' : [Branch[0]],

    'Designation' : [designation[index]],

    'Number' : [roll_no[index]],

    'Time' : [x.strftime("%c")],

    'Date' : [x.strftime("%b-%d-%Y")]

}

df = pd.DataFrame(data)

df.to_csv('G:/My Drive/projects/Attendance.csv',mode='a',index=False,header=False)
```