University of Essex

Department of Computer Science and Electronic Engineering

CE901: MS PROJECT AND DISSERTATION

BLOCKENDANCE : A Blockchain based Automated Attendance System

Mohammed Talib

Supervisor: **Panagiotis Kanellopoulos**

December 13, 2022

Colchester

ABSTRACT

An effective approach for managing personnel in corporations or institutions is the attendance management system. We can track attendance with the help of security cameras installed in every corner. And, by incorporating blockchain technology, we can make it more transparent and secure. Despite the growing use of blockchains for a variety of applications, it is still new in the education domain. In this work, we have shown how we can automate attendance monitoring and at the same time secure it. We designed and implemented an attendance monitoring system based on blockchain and computer vision (CV) technologies. This system is built with Python for the front-end and CV; for data, we have used mySQL Workbench. For the Web3 server, Node.js is used to integrate blockchain with the attendance management system. Major libraries used are OpenCV, Pillow, Tkinter, and mysql.connector. The system is designed to work on a real-time basis, which is why we have chosen light models like LBPH and HAAR Cascade. For file storage, Infura (the InterPlanetary File System) was taken into consideration, and for blockchain, we have chosen Ganache to deploy the chain in our local system. Accuracy of the model varied from 95-100% in different scenarios.

ACKNOWLEDGEMENT

This paper was completed with the assistance of many people, to whom I would like to express my heartfelt gratitude. Kanellopoulos, Panagiotis, my professor, deserves my heartfelt gratitude. His willingness to devote his time, suggestions, and advice is greatly appreciated. This paper would not have been possible without his guidance, critiques, and recommendations. I would like to express my gratitude to the University of Essex staff for always being available to help me, and for their assistance and support throughout my years of study. They have never failed to provide generosity and company when needed. Their efforts will not be forgotten. Above all, I'd like to thank my family in the UK for making this all possible. Thank you for your unending love, encouragement, and support. Every achievement is dedicated to you. A special thanks also goes to my dearest friend, Pankhuri Kulshreshtha, and Umer Chaudhary for helping with the computer vision and blockchain. Thank you for sticking by me the whole time.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Attendance monitoring with face recognition is a modern alternative to the outdated system of recording attendance. The suggested system is a Python-Tkinter-based system that uses a MySQL database for support. It is suggested that this system be built on biometrics, specifically face authentication. This approach totally eliminates the possibility of faking attendance, which is a concern with traditional approaches to attendance because biometrics are used. Every institute uses the attendance management procedure, which is important for keeping track of students' progress. Each institute approaches this in a different way. Some of their institutions still rely on outdated paper- or file-based systems, while others have introduced automated attendance tactics that employ biometric technology. A "face recognition system" is a piece of digital software that can identify or verify a person by comparing facial features. At the time of attendance, when the system camera turns on, the system will detect faces within the frame using HAAR Cascade, which is CV2's popular library for detection. We will be using and comparing the LBPH face recognizer with other models to get the best results. Later .csv file will be generated, and we will be making a new .csv file every day.

Now, we have resolved the problem of forging the attendance, let's focus on the security of the same and understand how and why blockchain chain technology is better than the traditional method of storing attendance. Fabrications in the education system are becoming more common in several nations. According to several earlier studies, graduation document forgery needs a new solution, and blockchain technology can be that option [67][3]. According to other studies, Blockchain technology is used to track operational activities in a university to secure data [63][8]. Due to the distributed [64], secure[51], and traceable[65] nature of Blockchain technology, this research attempted to replicate the use of multichain [8] to record student attendance. This issue will have a fresh solution thanks to blockchain [89, 2]. To stop the institution's graduation documents from being faked, for instance, by a fake attendance report. Numerous universities issue original degrees, but the student had never participated in academic activities, therefore the inconsistency was discovered [64]. Early studies noted a similar problem with faking attendance[72].

It is anticipated that blockchain technology will offer a solution to this issue. In order to gather feedback and validate the simulation outcome, this qualitative research involved nine universities, both private and publicome, this qualitative research involved nine universities, both private and public. The purpose of this study is to explore blockchain technology for storing student attendance. According to the research findings, computer vision can make it easier to track students' attendance during lectures, which is crucial for maintaining control and oversight over university operations, and for secutiry reason, the usage of blockchain technology makes it secure and uneditable . The limitations of the current system's capabilities, such as how difficult it is for users to update data But with blockchain technology, even the data may be recorded and will never be readily altered or changed, making it immutable. Beginning with an abstract, the paper is organised into an introduction, a review of the literature, a method, an implementation, a discussion of the results, and a conclusion.

CHAPTER 2

Literature Review

The purpose of taking attendance at schools, universities, and other types of organisations is to maintain a record of the number of students who are currently enrolled. It is a very significant thread in the process of maintaining discipline among students or employees in a university/business and giving excellent education in schools and universities, and if someone strays from needed standards, appropriate action can be done[77]. A method known as an "attendance system" is used to keep a record of a person's presence at a certain moment in time. According to the findings of a few different pieces of research, it was deduced that the academic success of students is correlated with the number of classes that they take[28][19].Because every college and university requires all of their students to be present for all of their classes, a variety of attendance tracking systems have been developed and are used across the sector.

Integrity and legitimacy of attendance are significantly essential, and there are various techniques used by different organizations. QR Code scanning is a technique for tracking attendance in which you use your smartphone to scan a QR Code present in a company and your attendance is marked. Individuals can misuse this system by giving out their credentials to someone else, and that person will mark attendance on their behalf, making it a less than reliable technique. Everyone in the organisation can be counted as present with only one quick scan of the QR code picture, which can easily be shared by anyone. Because of this problem, the Global Positioning System (GPS) system was implemented, requiring students to scan the QR Code upon entering the classroom in order to mark their attendance[25].One downside of this strategy was that cybersecurity threats were not identified. Another study proposed using QR Code generators to generate a new code for every class in the student account; lecturers would then scan these codes in place of GPS devices. However, the aforementioned method was criticised for being time-consuming and requiring a great deal of human effort[70]. On the other hand, a different approach involving AI and biometrics was offered. The biometric technology was faultless since it completely eradicated attendance forgery and had no security holes that would allow students to game the system. The main issue of 'proxy attendance' was also solved, as you cannot ask your friend to mark your attendance on your behalf[47]. With the present COVID-19 condition of social distancing and avoiding touching things as it can bring you into contact with germs, this finger print scanner attendance method is no longer viable. Since the coronavirus can live on a surface for up to 16 hours, biometric identification methods like fingerprints and RFID (radio frequency identification) chips on cards are avoided. With the concern of contaminants

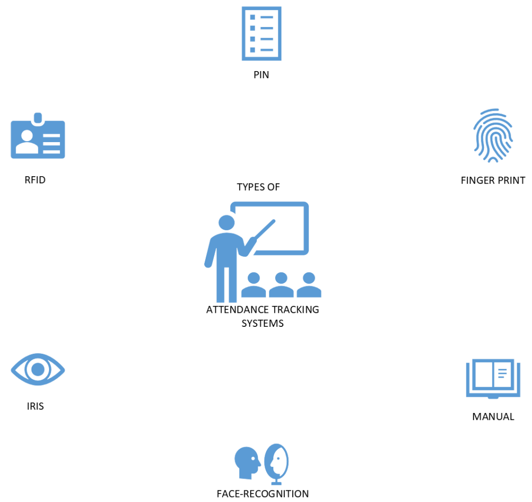


Figure 2.1: Types of Attendance Tracking System

in mind, a contact-less attendance system utilising closed-circuit television (CCTV) and artificial intelligence-based CV was implemented. This system permits marking attendance as soon as a student enters the classroom. It can detect a student's face just by scanning it; there is no need to stare into the camera or strike a specific position; it can automatically identify you even if you are in a crowd[11].

In the early 1970s, computer vision was viewed as the visual perception of films and images that mimicked human vision and intelligence[84][71].Â

Computer vision is sometimes characterised as giving a machine the ability to see, interpret, and extract information from images, videos, and other visual inputs[84].

2.1 Use of Computer Vision for Attendance Monitoring

The primary purpose of this study is to design a facial recognition-based intelligent attendance management system that can later take advantage of blockchain technology to store the data. This will solve the issue of attendance tampering and 'buddy punching' or 'proxy attendance'. Â The primary method is to compare a recent picture of a student with pictures of the same student that were taken on purpose and kept in a database; if the pictures match, the student's attendance will be marked as present. The model that was proposed Â by Naveed et al. [9] was connected to two databases in which one was for faces, while the other was used to record attendance. Prior to face detection and recognition, the camera is used to capture the student's facial image and remove background and noise from the image.

In a different implementation of a comparable system, Kawaguchi et al.[48] developed in which face recognition was used along with the location to know if the student is attending the right class. This is a continuous examination method that uses video streaming cameras to detect student attendance in the classroom. Moreover, they approximated the seating configuration using a variety of calculations. Â The model suggested by Marko Arsenovic et al. [7] Â employs the most recent developments; cascaded Convolutional Neural Networks for face identification and Convolutional Neural Networks for face embeddings. CNNs achieve the best results with larger datasets, or in other words, their primary test consisted of using these algorithms to small datasets. On a small dataset of the initial face images of workers in the present state, the overall precision was 95.02 percent.

Facial detection system put forth by Kruti Goyal et al. [27] is an algorithm-based facial detection model that uses multiple types of algorithms, such as AdaBoost and Haar Cascades. The model used MATLAB and OpenCV for it's construction . The extraction of facial characteristics is accomplished as a result of pattern recognition-based face localization.

In their study, Nusrat Mubin Ara et al[6] reviewed the recent progress in the field that they employed, including facial recognition, standardization, and face recognition. These are some of the topics that were covered. The authors also described the process of face identification using the history of oriented gradients, face alignment using facial landmark estimation, feature extraction using a convolutional neural network, and finally embedding generation.Despite the fact that their algorithm made some incorrect predictions, they attained an accuracy of greater than 95

Priyanka Wagh et al. [92] examined the success rates of several facial recognition systems, such as Principle Component Analysis, Eigenface, Neural Network, and Support Vector Machines. The authors also discussed system architecture, a step-by-step approach, and the algorithm that supported it. Additionally, they have created a mathematical model employing mathematical principles and language.

The statistical techniques PCA and LDA were used by Abhishek Jha et al. [45] to develop a better method for the recognition process. They also compared the taken and saved pictures to record attendance. They referred to the lengthy and error-prone process of involvement formation, which, if not negotiated, may have a significant effect on the student. They offered a framework for determining the images in accordance with a certain approach so that matches could be scored. While it may be possible to do this by the use of certain computations, such as colour detection, PCA, and LDA, there are likely more efficient methods. They extracted several facial elements from the image, including the face's structure, nose, and eyes, among others. Eigen Values are utilised by PDA and LDA in order to correctly record students' attendance.

2.2 Blockchain's potential in education system

Students, professors, employers, accreditors, validators, and testers are just a few of the characters involved in the use of educational system. Don & Alex Tapscott claim that blockchain techonology based innovations in higher education could be advantageous in the following cases [86]:

- Records and identity - All the student related information can be remain secured and authenticated.
- New Pedagogy - To uncover learning trends, enhance teaching, and create new learning models, the ledger might hold data on chosen courses and the network of learning interactions.
- Payments and incentives: determining the funding strategy and rewarding students for their accomplishments

- Meta University Model - to get knowledge of how to redesign the university to meet new demands from students and employers.

According to the author, the plus point of the distributed ledger system is "to record virtually everything of value and importance to humankind, starting with birth certificates and moving on to educational transcripts, social security cards, student loans, and anything else that can be expressed in code."[86].The blockchain can be used by employers or qualification validators to view official college transcripts. Additionally, Sharples et al. stressed that by utilising distributed ledger, "organisations and individuals could gain further reputation by providing recognised services to education, such as providing open courses, or funding research[76]" Additionally, as the ledger is open to the public, anybody can view how the said person "received that particular qualification, and the standards for adding additional value are agreed upon by consensus.[76]"[4].

MIT has recently introduces a new open platform to get the certificates under the name of 'Blockcerts'. The following briefly describes usecase linking MIT with students, employers, and others[4]: 1) Through the certificate wallet, the graduate and request to receive a certificate from the school. 2) The certificate is created and given to the graduate by the issuer; 2) The recipients can share the given certificate with the employer 4) If employer wants they can verify the certficate with the help of hash key provided.

Sony Global Education (SGE), already has created a technology that uses blockchain technology to the educational sector, "leveraging blockchain's secure properties to realize encrypted transmission of data - such as an individual's academic proficiency records and measures of progress - between two specified parties[24]". The SGE system, according to Georgopoulos (2017), is also based on IBM's HyperLedger cloud service and serves as a "centralized ledger for storing educational information, such as degrees, diplomas, tests and more, while preventing fraud[26]". University of Nicosia become the first higher education institute to issue the degrees using blockchain which can be later verified using the hash key[4].

2.3 Blockchain Vs Database

It should be obvious from the table 2.1 and figure 2.2 thatÂ that blockchain isn't a general-purpose technology and should only be used carefully in order to benefit from it. Business analysts and system architects can utilise the straightforward decision flow control model in 2.2to determine whether or not blockchain technology should be deployed. In general, application scenarios involving many administrative authorities and a lack of confidence between them benefit from blockchain technology. A familiar application would be a supply chain management system, in which numerous parties collaborate to deliver items.

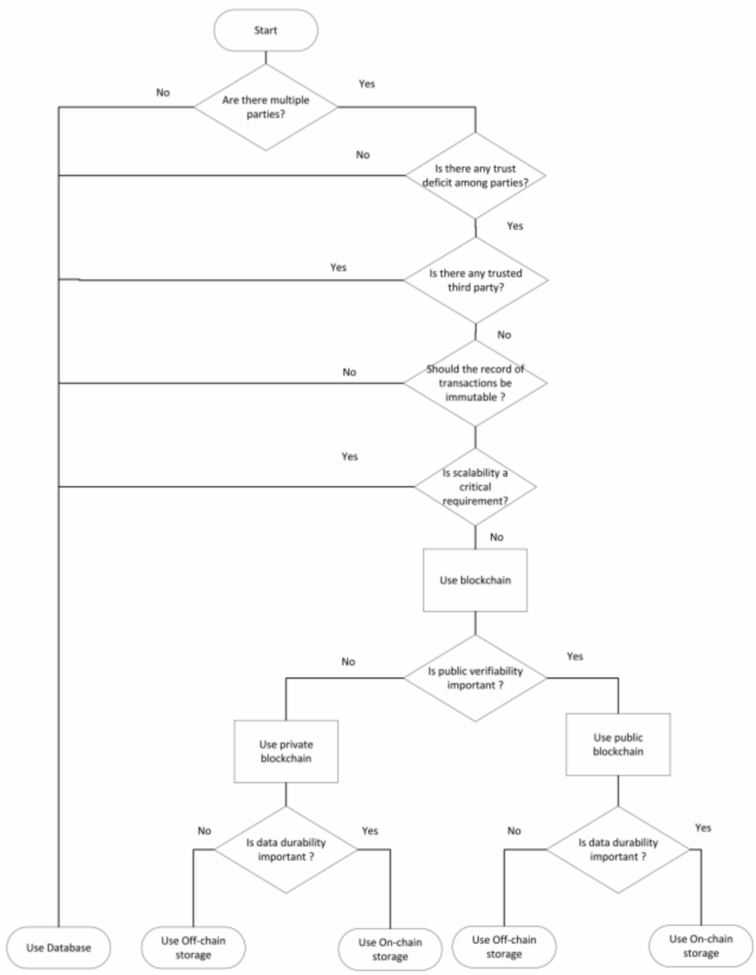


Figure 2.2: When Should We Use Blockchain?[15]

Another instance may be a group of independent businesses working together on a government contract where there would be a lack of confidence between the participants.

Usually, collaborating partners choose a third party that both parties can rely on when there is a trust gap between them. For instance, sellers and buyers trust the bank to handle their financial transactions. Finding a reliable third party can occasionally be difficult or dangerous. For such use-case, blockchain is a useful technology if all the aforementioned requirements are met. However, we must now choose between using a private blockchain and a public blockchain. A public blockchain should be used if the value contained in the chain needs to be publicly verifiable, whereas a private blockchain is preferable if the data is only for a small group of people because the consensus methods can be made simpler.

Issue	Blockchain	Central Database
Trust Building	No trust party is needed.	You must have a trust party
Confidentiality of Data	Transparency is available for everyone	Only authorized person can access the data
Robustness	Data is distributed among nodes	Data is stored in central database
Performance	For verification it takes a little time.	It approves or validate immediately
Redundancy	Every member of the chain has a latest copy of the transaction	Only one central party has a copy
Security	Uses cryptographic signatures and rules	Only uses traditional access control

Table 2.1: Comparison of Blockchain based database and Central Database[15].

CHAPTER 3

Theoretical Background

3.1 What is Computer Vision?

Computer vision is a sub field of artificial intelligence (AI) which allows machines and computers to extract relevant information from digital photos, videos, and other optical inputs, and then conduct actions or make suggestions based on that data. If artificial intelligence helps machines to think, computer vision allows them to see, analyze, and comprehend. Computer vision functions similarly to human vision, with the exception that humans have quite a kick start. Human vision has the benefit of a lifetime in context to learn how to distinguish between things, how far away things are, if they're in motion etc.

Deep learning, the dominant sub field of artificial intelligence (AI) , has indeed been augmented with diverse network architectures. DL could capture the characteristics of huge data effectively and automatically. Current DL applications include computer vision (CV) , natural language processing (NLP) , video/speech recognition (V/SP) , and banking and finance (F&B)[13].

In the early stages of computer vision research, the deep learning technique encounters challenges because to memory, CPU, and GPU restrictions. Thus, the majority of academics are investigating the application of machine learning in computer vision. K-means, Naive Bayes classifier, Decision Tree, Boosting, Random Forest, Haar Classifier, Expectation Maximization (EM) , K Nearest Neighbor (KNN) , and Support Vector Machine have all been presented as approaches for CV (SVM) . Viola and Jones (2001) [90] employed the Haar-like

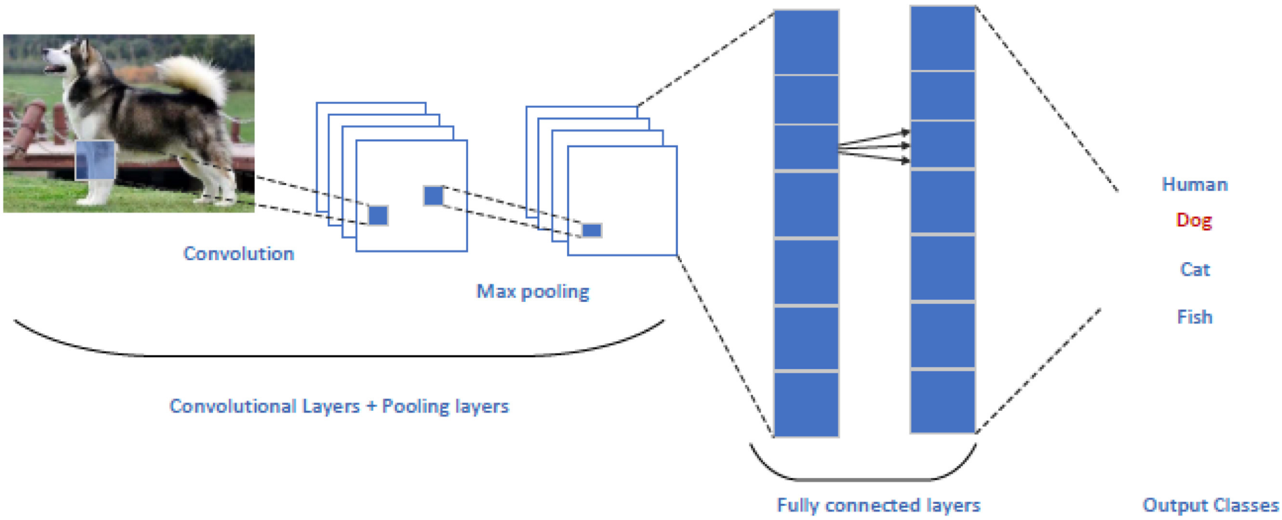


Figure 3.1: CNN architecture for image classification.

wavelet feature and integral graph technique for face detection built on the Adaboost. They weren't the first to offer wavelet features, but they have built more helpful face identification features and cascaded the powerful classifier trained with Adaboost. The algorithm proposed is known as the Viola Jones detector. OpenCV's Haar classifier was created by Lienhart and Maydt (2002) [54], who further enhanced this detector using rotated Haar-like features and created the Haar classifier.

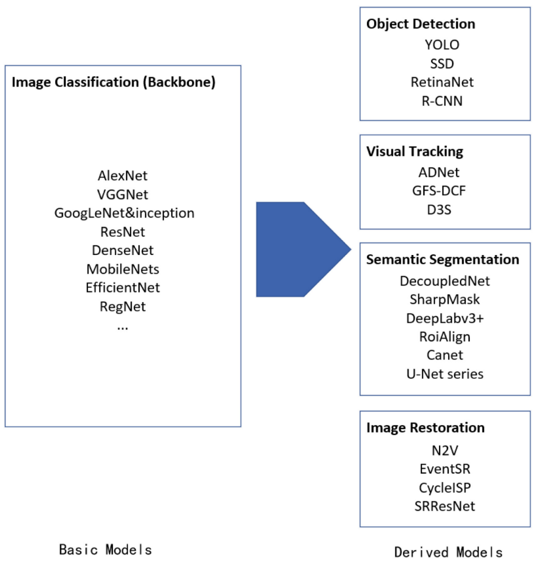


Figure 3.2: History of Computer Vision

Image classification constitutes one of the most essential and fundamental computer vision fields. The CNN architecture for image classification is depicted in Figure 1. Convolutional layers, pooling layers, and fully linked layers make up CNN. In the convolutional layers of a CNN, several kernels are used to convolve the entire image and intermediate feature maps, hence yielding multiple feature maps. Pooling layers are utilised to decrease the size of feature maps and network parameters. It serves as a CNN classifier at the conclusion of each CNN architecture's completely connected layers. After the completely connected layers, the output can be used for classification tasks, as depicted in Figure 1, or it can be transferred to the subsequent Deep Neural Networks (DNN), as depicted in Figure 2. Therefore, according to Szegedy et al. (2016)[83], research achievement advances in classification results tend to translate into considerable quality gains across a broad range of application sectors[13].

Furthermore, we highlight current breakthroughs in DL by examining the eight developing techniques that have emerged as the fundamental models in numerous CV application sectors, namely VGGNet, EfficientNets, ResNet, DenseNet, GoogleNet & Inception, MobileNets, AlexNet, and RegNet. The extensive uses of DL can be condensed into four primary application scenarios: detection, visual tracking, semantic segmentation, and image reconstruction. [13]

3.2 Face Detection

Face detection is a significant step in face recognition, which relates to the localisation of the face inside a picture. After detecting a face, it also is trimmed for recognition. In spite of the existence of a number of face detection systems prior to the year 2000, performance was not nearly sufficient. This was until the proposed approach by Viola Jones et al [91]. The Viola-Jones face detector is able to be constructed and run in real time due to three key concepts: the integral picture, classifier learning using AdaBoost, and the attentions cascade structure[44].

The integral image, also referred as a summed area table, is an approach for finding the summation of values in a rectangular subset of a grid fast and effectively. It is initially brought to the realm of computer graphics for usage in mipmaps. The integral image was utilised by the Viola-Jones face detector for the quick computing of Haar-like characteristics. The characteristics are described as the (weighted) difference in brightness between two to four rectangles. Boosting is a strategy for discovering a highly correct hypothesis by combining numerous moderately accurate hypotheses. Attentions cascade is an integral part of the Viola-Jones detector. The essential idea is that smaller, more efficient boosted classifiers that reject the majority of negative sub windows while retaining nearly all positive examples can be constructed. On the basis of testing on the MIT+CMU face dataset, it has been determined that this method is 15 times faster than that of any previous method[44].

The Viola and Jones approach is the very first to put rectangular boxes on the face. However, it has significant disadvantages due to its big feature size. The total amount of Haar like characteristics in a 24x24 image is 160 000, and it does not account for wild faces or frontal faces. As a result, numerous algorithms with more powerful face detection features, such as HOG, SIFT, SURF, and ACF are used. Normalized Pixel Difference (NPD) is a new feature introduced in [53] that separates the intensity of two pixels. Support vector machine (SVM) is used as a classifier in the well-known approach Dlib [50] to enhance the robustness of face identification [44].

Deep Learning has some of the most researched work, such as automatically learning and synthesising face-detection features [97]. A system for multitask learning that combines a ConvNet and a 3D mean face model. Using CNN cascade and the region proposal network, additional enhancements were made (RPN). In face identification benchmarks such as FDDB [43], faster R-CNN and ResNet [79] have led to considerable performance gains. One of the fundamental downsides of deep learning is that the information is encoded in a scattered fashion, making it difficult to comprehend [44].

3.3 Face Recognition for Digital Images

Face recognition for digital photos can be classified based on the employed feature, such as global appearance-based approaches and local feature-based methods. There are also approaches based on neural networks and Deep Learning among the remaining techniques. Figure 2.4 provides an overview of the facial recognition methods addressed in this study.

3.3.1 Global appearance based methods

These techniques are derived from the statistical representation of feature data. There's many ways to extract facial features from an image. Principal Component Analysis (PCA), kernel PCA (KPCA), 2D-Image PCA (2D-IPCA), and Fisher-Linear Discriminate Analysis (LDA), among others, are a few of these techniques. All of these techniques focus on extracting global features, also known as global features, from an image.

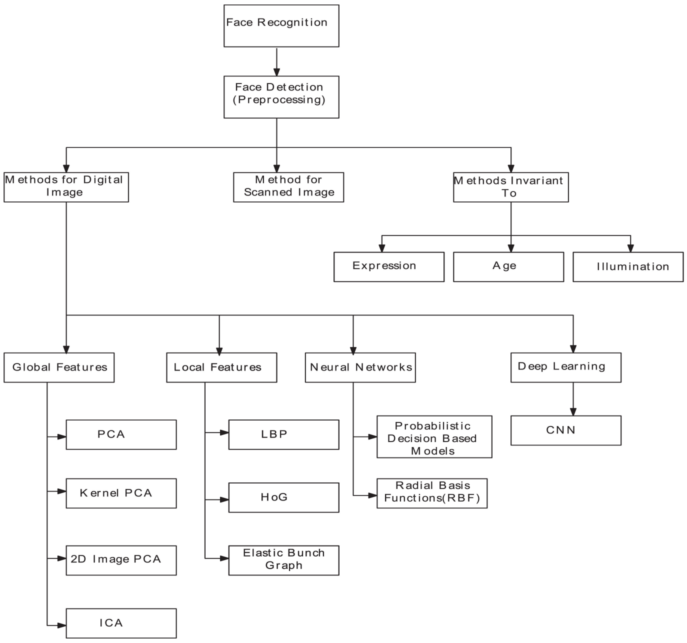


Figure 3.3: A description of the face recognition techniques

Principal component analysis

This technique relies on the Karhunen-Loeve Transform [78]. To begin, a 1D vector is created from the 2D face image matrix to stand in for a single column vector inside the dataframe. After transforming each facial image into a data matrix, the covariance matrix is calculated. There is a perfect coordinate system that may be used to represent a face, and we call these representations Eigen Pictures. The KL transformation was done to the face photos containing only the eyes and nose after they were cropped. A vector has been created from a collection of cropped face pictures that are referred to as a face ensemble. Here, only photographs of upright faces have been selected. To determine how far each face image is from the average, we first calculated the mean face vector. These face ensemble pictures undergoes PCA, and the highest eigenvalue corresponding Eigen vectors referred to as the Eigen picture[44].

Thereafter, the work was expanded in [88], in which it was considered a 2D face recognition issue. A computational model that is quick, straightforward, and accurate in a limited setting has been constructed. Eigen face, which analyzes faces overall, has prioritised an Information Theory-based method to encode the most pertinent information in a set of faces. Face pictures have been transformed from image space to face space using PCA. Both approaches have drawbacks, including an insufficient number of facial photos and the inability to function in unrestricted environments. Initial evaluations were conducted on locally accessible datasets, in which the accuracy was determined by adjusting direction, lighting, and size. The accuracy that has been recorded is approximately 64 percent [88][44].

Kernel principal component analysis

Using a suitable nonlinear mapping, the input picture is transferred to a feature space, and then PCA is done in the feature space. The method is known as kernel PCA . In face image analysis, the use of a polynomial kernel can offer a stronger correlation between input pixels. To recognise a person's face, features are extracted in the transformed domain and fed into linear support vector machines (SVMs). The approach has been evaluated on the Olivetti Research Laboratory (ORL) database and, with an accuracy of 83.3%, has demonstrated superior performance than a linear PCA [49].

Separate component analysis

PCA utilises the pairwise associations between the picture database's pixels. Important information is supplied in Independent Component Analysis (ICA) via the high-order correlations between pixels. It produces enhanced basis pictures for higher-order statistics. Two distinct architectures are presented in [10]. One of them treats images as random variables and their pixels as output, while the other does the opposite. Each row of the matrix representing the face database represents a different person. As stated previously, the first approach considers face images to be random variables and their pixels to be trials, and then calculates the degree of independence between two images. If two photos are independent, it would not be possible to anticipate the value of a pixel based on the value of the same pixel in another image while travelling across pixels. In contrast, the second method computes the independence of two pixels. Thus, it is not possible to forecast the value of a pixel based on the value of another pixel in the same image while travelling across it. The performance of ICA is enhanced when cosines are employed as the similarity metric instead of euclidean distance. It has been evaluated on the FERET database and achieved 86% accuracy [10], which is superior to PCA [78].

3.3.2 Locally based feature approaches

In contrary to global feature-based methods, local feature-based methods partition the facial images into many blocks and analyse each block individually when extracting features. The block that does not contribute significantly to distinguishing characteristics may be disregarded. Consequently, the accuracy of such approaches is superior. Finding appropriate descriptors for local face regions is difficult and remains an ongoing question. Any such descriptor should preferably be simple to compute and possess a large inter-class variance and a low intra-class variance[44].

Local binary pattern (LBP)

LBP has been proposed largely to characterise texture patterns [66]. The fact that the human face also consists of micro patterns encourages its use as a face descriptor. LBP gives a binary labeling to each pixel of an image via imposing a limit to its surrounding 3 x3 neighbourhood. Then, a texture description consisting of a histogram of the labels is produced[1]. Some facial characteristics (such as the eyes) contribute more to recognition of human faces than any others. The areas are weighted by their importance. Chi square distance [87]classifies similarity. This approach got accuracy of 95% on the FERET database[44].

Histogram of gradient orientation)

HOG is 2D rotation- and scaling-invariant[16][59]. The image is first separated into small connected areas, or cells, and a histogram of edge orientations is produced for each cell. Normalizing histogram counts yields the HOG description [18]. The method's accuracy on FERET is 95.5%[44].

3.3.3 Methods supported by neural networks

As imagery and storage technology progresses, we witness a tremendous increase in the database's size and quantity of images. Conventional image processing-based statistical methods [81], [80], [40], [35], [37], [39], [98], [57]fail to achieve space-time efficiency in these situations. Listed below are some neural network-based facial recognition techniques[44].

Probability-based decision-based neural networks)

This technique [56], uses two levels of facial recognition. Nose and eyes give distinctive facial traits that are invariant to facial expressions, hairstyles, and mouth movement. The secondary facial features, such as mouth and hairline features, are produced by normalising and reducing the resolution of the input face images using intensity and edge values. Two neural networks based on Probabilistic Decision Theory are fed with feature vectors (PDBNN). On FERET and ORL, the technique achieved 84% and 98%accuracy respectively[44][96].

Radial basis function (RBF)

RBF is a neuron classifier that is beneficial when the training data setÂ is small and the face images have high dimensions. PCA and Fisher′s linear discriminant (FLD) [94]are applied to facial characteristics to provide lower-dimensional discriminant features. An algorithm for learning trains an RBF neural network. PCA and FLD generate most discriminant features so that training pattern classes can be splitted far[44].

3.3.4 Methods that leverage deep learning

Prepossessing, feature transformation and, local descriptor extraction are all steps in the conventional approach to face recognition. Despite the fact that these processes have been enhanced independently, face recognition accuracy has not increased significantly. In addition, the majority of approaches for unrestricted face recognition only address one of three challenges: position, expression, and illumination. Therefore, traditional approaches cannot extract real-world invariant features for recognition [62]. However, deep learning approaches learn numerous representations in a stack of layers. The retrieved features from these layers are adaptable to variations in lighting, emotion, and position. Using With the help of neural network [30],[29],[38],[36],[33],[99],[31],[29],[34],[23] and,[100], these methods stake one beneficial representation above others. The features obtained from a deep neural network's several stacked layers demonstrate robustness toward any form of change in general[44].

DeepFace is the name of the first deep learning architecture proposed in [85]. It employs a nine-layered CNN with completely connected last two levels. The completely connected layer′s output is delivered to a K-way soft max function. Soft max layer outputs class probability distributions. The accuracy of this method's verification on the LFW database was 97.35 percent[44].

Another strategy based on learning a euclidean embedding for each image with CNN was proposed in [75]. The training of the network is done in such a way that comparable face representations have a smaller L2 distance than different faces. It presents an online triplet mining technique that makes use of roughly aligned triplets of both matched and non-matched facial photos. This approach has 99.63% accuracy on LFW. Euclidean and Softmax loss margin-based losses are incompatible, according to Liu et al[58]. Instead of taking into account the Euclidean distance, the authors recommended an angular margin. Modified softmax loss learns featured from angular features. This improved approach was 97.88% accurate on LFW. In [17], a new loss function called the "additive angular margin loss function" was made to make it even easier to tell one class from another. This network, ArcNet, has 99.83% LFW dataset verification accuracy[44].

[93]proposes light CNN framework which helps to learn the facial representation fromÂ noisy large-scale data. Rather than using ReLU, this method uses Max-FeatureMax operation to get a small representation and separate the informative labels from the noisy ones. On the LFW database, this approach has achieved 97.50% precision. [60]proposes a CNN-based facial recognition model with one trunk and two branch networks. The trunk network acquires knowledge of the features by studying a variety of face photos of varying resolutions. Resolution-specific coupled mappings(CM) are learned by tiny branch networks. The CM projects the high-resolution database photos and the low-resolution probe images into a space wherein their distance is minimised. This method's face verification accuracy on LFW dataset is 98.7% when probing image is 112 x 96[44].

In contrast to the majority of deep learning models, which conduct frontalization or normalisation of face pictures in order to deal with pose variation, in [61] a model was developed that analyses numerous face images by utilising several pose-specific CNN models. This model was developed to tackle the problem of face variation. This approach finds landmarks in all photos and aligns them to five reference poses. In each position, CNN models extract features. These were trained with 500,000 face photos from the CASIA web face dataset. The IJB-A dataset was used to evaluate this method, and the results showed that it attained a recognition accuracy of 93.1% for top-5 matches and 86.2% for top-1 matches.

S. No	Method	Benchmark Databases	Accuracy
1.	LBP	FERET	95%
2.	HOG	FERET,AR,CMU Multi PIE and YALE	95.5
3.	EBGM	FERET	89%
4.	PCA	LOCAL DATASET	64-96%
5.	ICA	FERET	44-86%
6.	kPCA	ORL	83.3%
7	2DPCA	ORL, YALE and AR	84%
8.	PDBNN	ORL, FERET, and LOCAL DATABASE	84-98%
9.	RBF	ORL	98%
10.	CNN	LFW	97.35-99.83%

Table 3.1: Popular Face Recognition Techniques[44].

3.4 What is Blockchain?

Blockchain is a novel technology built on hashing that serves as the backbone of exchanges for digital currencies and the software that powers smart contracts.Â Satoshi Nakamoto initially unveiled the blockchain in his article "Bitcoin: A Peer-to-Peer Electronic Cash System", which set out the bitcoin's mathematical foundation. Despite this being a ground-breaking paper, the author's real name is unknown and it was never submitted to a standard peer-reviewed journal. Blockchain is not just the basis of all cryptocurrencies, but it has also found widespread use in the more conventional banking sector.[20]

Bitcoin was build on Blockchain 1.0 to eliminate the requirement for a third party in digital transactions and to prevent the practise of double spending digital currency, it resolved the two issues that have persisted for a long time:

Transaction Management

Financial institutions have relied on third party companies to execute electronic payments for a very long time. These third parties manage financial transfer between merchants and their customers. These third parties give additional client information and reverse transactions if needed. But, because there's no other way of making transactions over a communication channel that does not need a trusted third party. This can lead to an increase in the cost of each transaction and a restriction on the number of transactions that can be processed for a merchant in a given period of time. Blockchain technology allows interested parties to conduct transactions directly without the need for a preexisting third party. Instead of a trusted party, cryptography protects stakeholders' money [74].

Double Spending Problems

Digital payment system has existed for a long time, but it has always had the issue of duplicate spending. The potential for the same digital money to be spent more than once to conduct several transactions constitutes the double spending problem, which is a potential weakness in online payment systems. This is done by duplicating files or falsifying information to save money [82].

Blockchain is utilised in cryptocurrencies as a public ledger to record all transactions. In Blockchain, transactions are maintained as a data structure known as Blocks. As transactions take place, new blocks are regularly added, thereby extending the Blockchain.

Cryptocurrencies are regarded as the first use of Blockchain and are now operational as an Internet-based digital payment system. With the potential to programme cryptocurrencies as a decentralised trade network for all resources, Blockchain 2.0 had already been implemented to take advantage of the enhanced functionality of digital currency[82].

Blockchain 2.0, often known as "Smart Contracts," was the next major step in the evolution of the Blockchain sector. It is a concept for the decentralisation of markets generally and support for the transfer of many different kinds of assets, aside from digital currency, such as bonds, loans, mortgages, equities, smart properties, etc. [82]. It's created in such a method that automatically enforcing the rules that had been agreed upon by interested parties, similar to how standard corporate contracts function. With the development of technology, it became apparent that Blockchain might transform all industries, not simply markets, payments, financial services, and economies. This given rise to Blockchain 3.0, also known as Blockchain Applications, which extend beyond financial markets to include government, health, literature, and other fields [82].

Beyond financial markets, blockchain 3.0 provides a platform for creating distributed and secure applications. Through its connection with the technology of the web, it enables a universal as well as a worldwide scope and scale. It is viewed as a platform that will contribute to the creation of the "Smart World," particularly in terms of resource allocation for physical and human assets [5][95].

3.5 Types of Blockchain

Blockchains are divided into different types based on how they are used and what characteristics they have.

Public Blockchain

In public blockchain, you can join the network without any restriction or permission. Anyone can access the chain[12]. People can participate in the consensus process, view and submit transactions, and manage the public ledger. All users can publish, access, and validate newly generated blocks, allowing them to preserve a complete copy of the Blockchain[5].

Development and use of public blockchains are secure. Although any person can access the network and add ledger as blocks, the verification of these blocks requires computationally expensive consensus mechanisms such as solving puzzles or stacking one's own bitcoin. The modification of block contents is prevented by hashes and decentralised consensus. Additionally, a significant number of Blockchain nodes might be anonymous to safeguard their privacy[5].

In addition to their obvious advantages, public blockchains also provide numerous open research questions. Large numbers of participants and expensive consensus mechanisms make efficiency difficult [12].

Private Blockchain

Private blockchains are intended for use by a single entity. Members are invited to participate in the network and are responsible with maintaining the Blockchain in a decentralised fashion [12]. Private blockchains differ from public blockchains in that only authorised entities may join the network and update the blocks[5].

Private blockchains are seen more safe and economical than public blockchains because only known users may participate in the network while manipulation is similarly secured via hashes & approval of participants. Nodes on private blockchains are not, however, anonymous [5]. The open research challenges in private blockchains concern the altering of blocks and the hacking of the network by authorised internal participants.

Consortium Blockchain

Consortium blockchain is a special type of private blockchain that is designed to serve a number of different companies. Only authorized and trusted members are permitted to join and administer the network. The process of reaching consensus in a blockchain of this type is quicker than in public blockchains, although being quite slow in comparison to private blockchains [12]. Consortium blockchains are more secure than private blockchains in case of altering data. Hacking is also better protected in this sort of Blockchain due to improved security measures implemented by numerous organisations' members [12].

3.6 First Implementation of a blockchain.

The first blockchain application was based on the Bitcoin, with main data elements such as transactions and blocks stored in an encoded distributed ledger. There are two sets of transactions in which one includes transactions that are still in progress and the other contains transactions that have been completed but have not yet been added to the main chain. First, an origin(genesis) block was made. A timestamped collection of transactions are aggregated into each block and added to the distributed ledger. The process of "mining," which validates new transactions with significant computing cost, results in the creation of new blocks[4]. When any new transaction takes places, there are three blocks which are taken into consideration. First, the first block which helps to verify the previous transactions. Second is valid block, which contains all the encrypted information about the transaction and will join the chain one it is validated. Third block is called orphan block, by the name it is self-explanatory that it is not been accepted by the chain. When a transaction is taking place, there are lots of miners who helps to validate the transaction. However, at time when two different miners validates the transaction at the same time then for the time being both the blocks are accepted and later on when the chain verifies the chain with more blocks it discards the block of the smaller blockchain. Blockchain is designed in such a way that it can not be reversed, hacked and double spend[4]t.

Iansiti & Lakhani quoted by using BCT, "the ledger is replicated in a large number of identical databases, each hosted and maintained by an interested party"[52][41]. This distributed ledger is being verified and updated constantly in every transactions. Below are the reason why Blockchain or distributed techonology is better[32]:

- There is no central authority or server who has a control over the ledger.
- All participants abide by the ledger rules.
- Consensus among participants is used to validate a transaction.
- Smart contracts contain rules for transactions
- Digital signatures are used to sign and encrypt transactions on the ledger
- The ledger shows the progression of all transactions over time.
- Internet-based peer-to-peer connections are used for networks

"Miners" or "validators" are the contributors in the transaction verification process. They provide their computational power in exchange of earning digital money which can the bitcoin, ehtereum or any other coin.

The most common consensus procedures come in four types[32]:

- POW - Proof of Work, Contributors are needed to include transactions in a single block as miners, and they must then build a hash function using additional parameters.
- POS - Proof of Stake, Participants are required to have a stake in the network to serve as validators, and "the author of a new block is picked in a deterministic way," depending on that stake or wealth.
- XRP - Ripple protocol, The following recent closed ledger state includes transactions that receive 80% of yes votes.
- PoET - Proof of Elapsed Time, uses timestamps cryptographically signed by hardware.

CHAPTER 4

Requirements

4.1 User Requirements

- The Blockendance programme ought to recognise students and instructors correctly.
- To identify the student, the data (images of students' faces) must be processed appropriately.
- Once the attendance is marked, it should store in a local database as well as create a CSV file for the same
- Once the attendance is marked it should upload on a blockchain.
- Once the attendance is marked it should send the user a notification that student's attendance is marked.

4.2 System Requirements

The following is a list of the components that were utilised and necessary throughout the development and implementation of the system.

4.2.1 Node.js

Node.js is an open-source platform with extensive libraries of JavaScript modules that made the development of web applications easy.

4.2.2 NPM

NPM is a package manager which is used to manage the modules and packages that are used with node.js.

4.2.3 VS Code

Visual Studio Code is a free code editor used for the development of whole applications without the need for additional editors. It was created by Microsoft and includes a vast number of extensions. One can install extensions to the software via the Visual Studio marketplace. In addition to code completion suggestions and syntax highlighting, this programme has other functions.

4.2.4 MetaMask

MetaMask is a digital currency wallet used to store Ethers. Using MetaMask, we can also communicate with the Ethereum blockchain. MetaMask is available as a mobile application and a browser plugin, both of which may be used to communicate with decentralised apps. MetaMask enables users to access several networks using the same public address, as well as interact with private blockchains and local blockchains.

4.2.5 Truffle

Truffle is an open source smart contract creation, deployment, and testing environment based on Ethereum.

4.2.6 Ganache

Ganache is a private blockchain network for creating distributed software. Ganache is made up of a large number of Ethereum accounts that have been credited with test ethers. These ethers will be used to test the operation of the application.

CHAPTER 5

Methodology

5.1 Face Detection

Face detection is the first and most crucial step in performing facial recognition. Therefore, we will attempt to comprehend face detection before moving on to face recognition.

5.1.1 HAAR CASCADE

Haar cascade is an object detection method based on machine learning. In this instance, a large number of images are utilised in order to train the cascade function. It was proposed in 2001 by Paul Viola and Michael Jones and is regarded as the most effective object detection method [69].

Prior to data operations, the algorithm must be trained with both images containing a face and without a face[22]. Following successful training, image features are extracted. After that, the size of the target image is chosen. The training image is then placed on the target image so that the average value of a pixel in each section can be calculated. Now, a match is considered if the average threshold values are met [21]. Since one classifier is unlikely to be both sufficient and accurate, multiple classifiers are typically employed during image training [46]. The characteristics of a haar cascade are listed below:

Several additional classifiers are utilised to strengthen the classifier. ADA BOOST is an algorithm for machine learning that selects the most suitable match for a target image based on tests conducted on selected images by multiple classifiers. It may even be necessary to

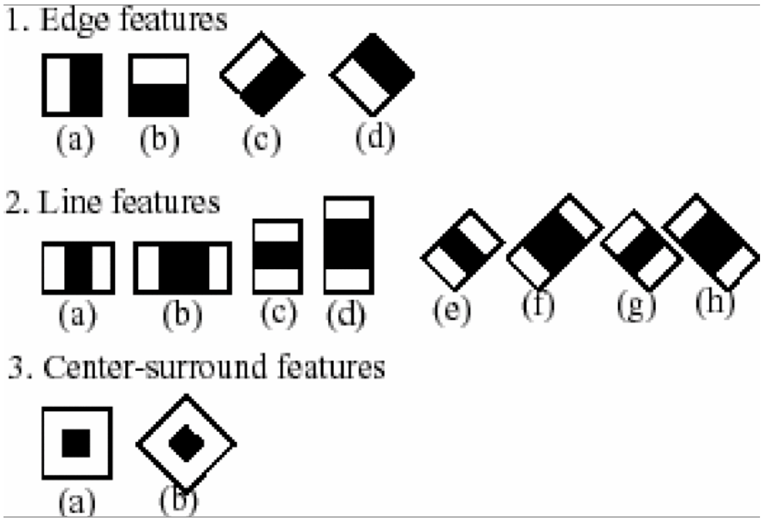


Figure 5.1: Haar Features[42]

reverse the process for better results It may even be necessary to reverse the process for better results

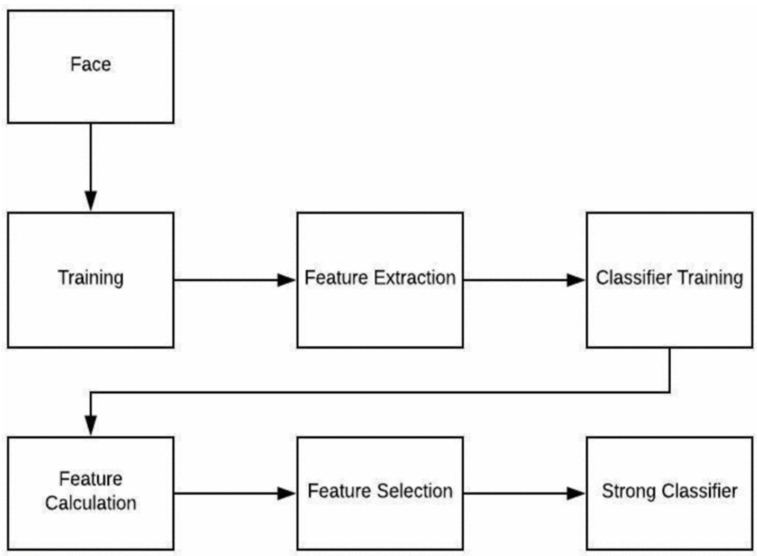


Figure 5.2: Flowchart for haar cascade[42]

For the purpose of visual recognition, the rectangle combination is used. While not technically haar wavelets, the rectangles in figure 3 can be classified as haar features. Haar feature is found by subtracting the dark-light region average. If the difference exceeds the limit, haar features are considered as being available[42].

Violas and Jones utilised a method known as integral image. Integral images are used to confirm the presence or absence of thousands of haar features. After extensive training, we can determine the presence or absence of a face in an image.

Figure 5.3: Haar-like characteristics matched with the face[42]

5.2 Face Recognition

5.2.1 LBPH

It's no exaggeration to say that LBP is the most productive texture operator currently available. It stores the centre pixel's value along with the threshold value of every neighbouring pixel. It takes into account the binary number form of the result [21]. The simplicity and discriminative power of LBP are what make it a more widely used method in many applications. Since its initial description in 1994, LBP appears to have improved as a texture categorization technique [46]. Later, it was found that LBP performs better on the same dataset when paired with histograms of oriented gradient descriptor. LBP offers additional features including computational simplicity and steady grey-scale shifts that enable real-time image analysis.

Let's examine the LBPH method in more detail. This algorithm has a number of phases. The first are the parameters, which are divided into four categories: the radius, the neighbours, the Grid X, and the Grid Y [55]. The image's centre is located using a radius. We'll refer to it as a local binary pattern for purposes of calculation, with a default value of 1. Neighbors, a collection of sample points used to create a circular binary pattern, are limited to 8 pixels due to potential computing costs. Grid X indicates the number of cells present horizontally, while grid y indicates the number of cells present vertically. With regard to the cost of computing power, grids x and y are both set to 8. The dataset must then be trained for the method. Every user has a distinct identity, and each image is kept with a distinct identity value [73].

The next phase in the computational process of LBPH involves the generation of an

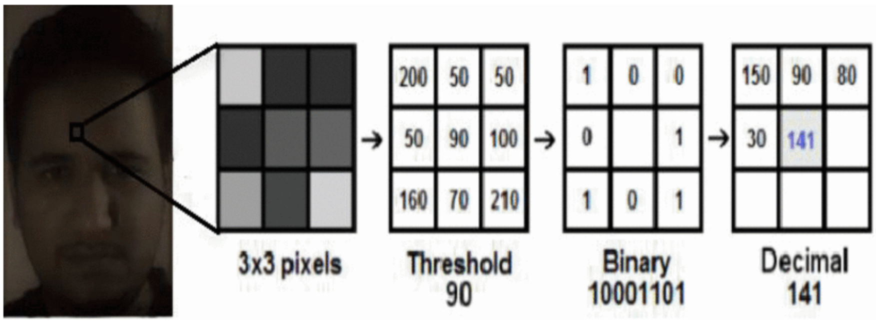


Figure 5.4: Gray scale value calculation for sub-matrix[42]

intermediary picture. This image utilises face traits to provide a more accurate description of the original image. Therefore, LBPH employs a sliding window determined by characteristics such as radius and neighbours. Let us have a grayscale picture of a person's face. We may obtain a portion of the image as a 3x3 matrix. Its intensity ranges between 0 and 255 per pixel. The value in the middle of the matrix serves as a threshold, and this number is what is utilised to establish new values based on the matrices' eight surrounding values [46]. Updated value of the matrix will be calculated by the central value which is . Now, if the value is more than threshold then it will be considered as 1 else it will be updated as 0 Further, the values are converted into into decimals and stored in centre and a new image is created with the help of updated values, which said give more characterstics about the image[55].

Using the picture created in the previous step, we divide it into different grids using the Grid X and Grid Y parameters. Because the picture has been converted to grayscale, each histogram will contain 256 locations, with each position denoting the frequency with which each pixel's intensity occurs. The histograms are then concatenated to form a new, larger histogram. Consider that if we have a grid that is 6 by 6, the resulting histogram will contain 9,216 places since 6 by 6 times 256 is 9216. The resultant histogram should accurately represent the features of the original picture. The final phase is facial recognition, for which the algorithm has already been trained. Histograms are used to represent each image in the training dataset. In order to generate a histogram that accurately depicts each new image, the procedures outlined above must first be completed. The image's histogram is compared to those of other images, and a match is determined based on the histogram with the most similar values. The returned result is the picture linked with this histogram. There are couple of ways in which you can calculate the difference between two histogram like chi-square, absolute value, Euclidean distance etc. In our situation, Euclidean distance was calculated using the following formula :

$$D = \sqrt{\sum_{i=1}^{n}(Hist1_i - Hist2_i)^2} \quad (1)$$

The algorithm's output is the identification of the image that has the closest histogram. In addition to returning the estimated distance, the method also produces a confidence value [73]. Keep in mind that a lower confidence score is preferable since it indicates a greater gap between the two histograms. The shorter the distance, the better.

By comparing the threshold to the confidence value, we can then determine whether or not the algorithm has correctly identified a picture. Now we can understand how the function of facial recognition operates:

cv2.face.LBPHFaceRecognizer_create()

- Using the parameter radius, a local binary pattern is derived from the centre pixel.
- Number of X and Y axis cells to be created.
- It will return -1 if the face recognition threshold is exceeded, indicating that the input face was not recognised [21].

5.2.2 Fischerface

Eigenface is the central idea underlying the application of the fisherface approach. The Linear Discriminant Analysis method is utilised for pattern recognition and serves as the foundation for fisherface, as depicted in figure: 5.5. While Principal Component Analysis seeks the maximum variance in the matrix, Label-Based Analysis (LDA) is based on the concept of labels so that the desired dimension can be projected onto the image. It is possible to maximise the difference between the means of classes [21].

The key idea is to get the highest possible ratio between the class scatter and its matrices, and LDA is utilised in order to accomplish this goal. We are aware that LDA functions more effectively in a variety of lighting circumstances compared to the PCA technique. On the one hand, Ronald Fisher's Linear Discriminant approach, which is based on pattern recognition, is applied to the variance of the image, yet , the fisherface technique, which builds on LDA, maximises the mean distance between classes and minimises variance. Both of these

Figure 5.5: Pattern recognition image[42]

techniques were developed by Ronald Fisher. Because of this, LDA is able to distinguish between feature classes more effectively than PCA and uses less space, making it a superior method [21]. Now we can understand how the function of facial recognition operates:

cv2.face.createFisherFaceFaceRecognizer()

- LDA utilises a variety of components to generate a fisherface.
- Like eigenface, it returns -1 if the threshold is exceeded.

5.2.3 Eigenface

The process of detecting faces is the most essential component of any face recognition system. Creating meaning from a number of input formats, such as photographs and videos, is a crucial component of this job. The data are typically of a noisy nature, which is a huge challenge as it can be disturbed by lighting or angle.Â Images are selected at random, but only if they contain a face; this is not a true random selection. We call the unique part of a face a "eigenface." Eigenfaces are made up of things like the nose, eyes, and mouth, as well as the distances between them [68].

The analytical method PCA is utilised by the eigenface algorithm. Combining eigenfaces allows it to recreate and reconstruct any original picture from the training set. Therefore, recreating an original face is only possible if a face is added proportionately. Input pictures should match learned images in pixels and grayscale.

Consider, for instance, the picture of the a x aÂ pixel. Each row is concatenated to generate a vector resulting in a 1xa2 matrix. The matrix created by averaging all of the photos from the training set.

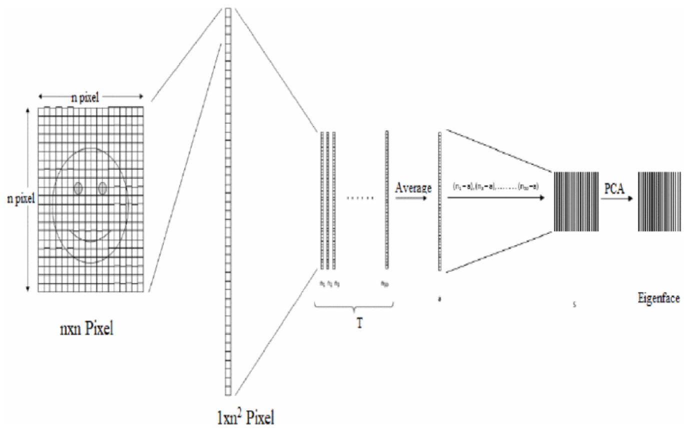


Figure 5.6: PCA calculation on the image[42]

After computing the unique feature, we next took each picture vector and subtracted the average face from it. The generated matrix depicts the average human facial attribute. This gives us the information we need to construct the covariance matrix. In eigen analysis, PCA is a useful tool. The result is expressed as a covariance matrix [21]. The first eigenvector has the most variance, the second eigenvector points in the direction of the third eigenvector, which is 90 degrees from the first, and so on. Each column is seen as an image, a duplicate face, which is known as eigenface.

In order to determine weights, the extracted features are projected onto eigenfaces. After importing and resizing the picture to fit the test data, the weights are used to compare the input image's characteristics to those from the other image sets. By comparing the input image to the entire dataset, we can identify it as a face. When it further compared it to each subgroup, we will be able to determine to which individual it belongs. We are able to utilise a threshold in order to manage the detection and identification processes, hence preventing errors caused by incorrect detection and recognition. Now we can understand how the function of facial recognition operates:

cv2.face.createEigenFaceFaceRecognizer()

- PCA uses the number of components to generate eigenface. According to OpenCV documentation, 80 pictures are recommended for optimal results.
- Just as Local Binary Pattern, it will also return -1 [21] when the threshold value for facial recognition is determined.

5.3 Blockchain

This section details the technological solutions and APIs used. This part is comprised of Smart Contracts, Solidity, Web3, Metamask, IPFS, and Ipfs infura. Intelligent contracts.

5.3.1 Smart Contracts

Smart contracts are small pieces of code designed to be kept in the distributed ledger. When the requirements and terms of the smart contract are met, they are instantly gets activated. At the level of the system, smart contracts are programmes that are executed as they were intended to be implemented by the developer or company that created them. The most significant benefit of using a smart contract is in the realm of blockchain technology, in which a completed transaction automatically generates a predetermined outcome without the need for any human intervention. Additionally, it makes it easier for everyone to understand one another and increases security.

5.3.2 Function of Smart Contracts

On a blockchain, the purpose of a smart contract is to facilitate the conduct of commerce and trade between unidentified and identified parties, and in some cases to eliminate the requirement for a third party to mediate the transaction.A smart contract eliminates the need for formalities and the related expenditures that are involved with using traditional techniques, all without sacrificing the legitimacy and credibility of the transaction.

5.3.3 How do Smart Contracts Work?

LBPH! (**LBPH!**) The conditions "if/when...then..." are included in the code of smart contracts that are kept on the blockchain, and they are used to ensure that smart contracts adhere to the fundamental concepts that underlie them. After the aforementioned conditions have been satisfied and confirmed, the activities are carried out by a network of computers that are also referred to as nodes or miners. Upon completion of the transaction, the blockchain is updated.

5.3.4 Benefits of Smart Contract

The advantages of smart contracts and blockchain go hand in hand.

- Safety: All of the transactions that take place on the blockchain are encoded with the public key and the private key, and the blockchain also returns everything with hash values, making it extremely difficult to hack. On a distributed ledger, every record is linked to the block that came before it and the records that come after them; hence, the entire chain would need to be modified in order to change a single record, which is not conceivable even with the help of supercomputers.
- Trust: Smart contracts conduct transactions in accordance with specified rules, and parties share encrypted records of those transactions. Thus, no one needs to wonder if information was edited for personal gain.
- Ease of use and accuracy because smart contracts are entirely digital and involve an automated procedure; as a result, no time is wasted and the outcomes cannot be altered in any way. Likewise, computer code is more precise than the legalese used in conventional contracts.

5.3.5 Solidity

Solidity is a high-level programming language that is based on the concept of contracts. On top of the Ethereum block, Solidity is used to construct smart contracts for decentralised application programming interfaces (DAPPs). The syntax of solidity is highly comparable to those of other scripting languages. Solidity is designed as a programming language to improve the Ethereum Virtual Machine. Solidity is a scripting language that is statically typed, which means that the constraints are checked and enforced at compile time instead of run time. C++, Python, and JavaScript are all inspirations for the Solidity programming language.

5.3.6 Web3.js

A group of built-in libraries known as Web3.js enable communication with nearby or distant Ethereum nodes via HTTP or Interprocess communication (IPC). Web3 is built for client- and server-side functionality. As an intermediary between the Ethereum blockchain and the smart contract, web3.js can be thought of as a gateway. This is undoubtedly the most cutting-edge JavaScript library out there.

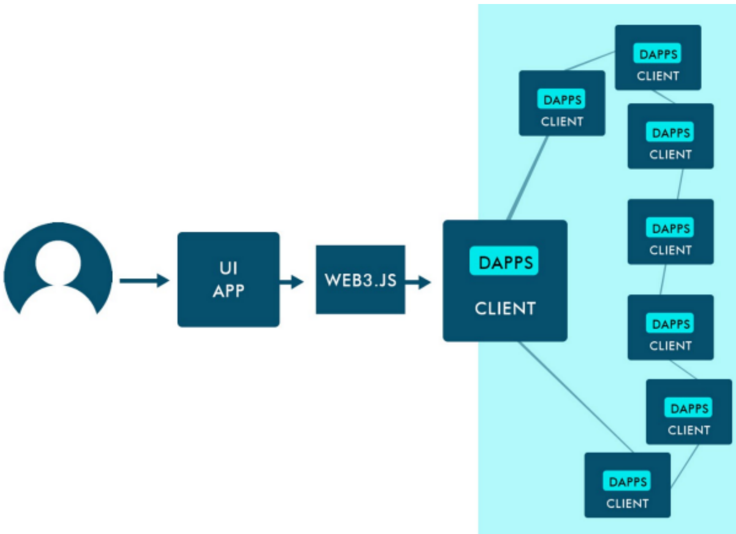


Figure 5.7: Web3.js Backend

5.3.7 Metamask

MetaMask is a chrome extension compatible with Windows, Mac OS X, and Linux. It is of use to us in the process of setting up and testing the Ethereum wallet. After being installed, it gives user the ability to hold Ether, which enables them to conduct transactions to any address that accepts Ethereum. MetaMask facilitates the execution of decentralised applications (dApps) without being a node on the public Ethereum network.

5.3.8 IPFS

Keeping documents or even just some data on a blockchain can be an extremely time-consuming and expensive process. Moreover, the process will be quite slow. Consider the issue posed by the fact that blockchain only supports short strings, such as the hash that contains transaction information. How will we store huge files or photos on the blockchain? This is where IPFS assists us in storing huge files and protecting them with Blockchain.

Interplanetary File System is sometimes known as IPFS. It is also acknowledged as a versioned file system that is comparable to git in that it can store files and track versions. The other component of how files are saved on a storage system comparable to BitTorrent. Combining these two ideas enables IPFS to create a new decentralised storage web that may outperform the current HTTP web.

IPFS always builds a distributed storage web that is permanent. In contrast to HTTP's location-based scheme, IPFS utilises a content-based addressing system stored on the network.

- A link of HTTP request looks like http://1.2.3.4/attendance/markattendace.csv
- A Link of IPFS request appears as/ipfs/Qma1MaVQ1b5R/attendance/markattendace.csv

IPFS employs the representation of material rather than a hard-coded connection location. The IPFS network stores files using their cryptographic hashes as addresses. The hash indicates the root block, node, or tree, and when the file is traversed, further blocks can be discovered. IPFS provides access to the beginning of the data, which aids in locating other data segments. In contrast to the HTTP web, where files and websites may be accessed from servers located anywhere in the globe, IPFS stores all data locally. I do not have to connect to the central server if someone close to me already has what I'm looking for; instead, I'll receive it straight from them. Without navigating to a specific URL, IPFS locates certain files directly. IPFS does this by combining a number of proven peer-to-peer system concepts.

Distributed Hash Table(DHT) is used to store IPFS data. Using the hash, we request a peer carrying the specified content. Using BitTorrent-like methods, the Data is exchanged between the network's nodes. For cases like request for a particular piece of content, IPFS will locate the node in its immediate neighbourhood that has access to that resource. In addition, IPFS will use Merkle Tree.

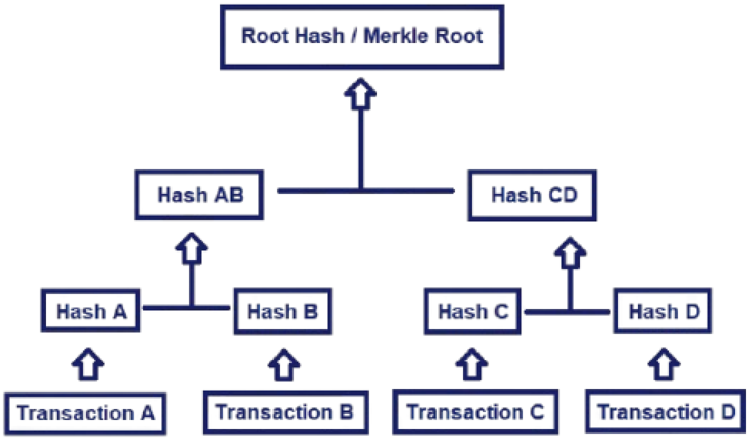


Figure 5.8: Merkel Tree

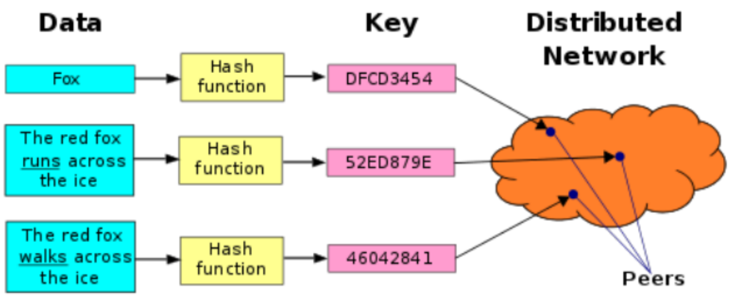


Figure 5.9: Merkel Tree

5.3.9 Infura

Infura is a platform which help you to interact with the Ethereum Blockchain. Infura effectively gives the required tools for any application to begin building on Ethereum instantly, instead of running the complex infrastructure themselves. Infura facilitates IPFS network connectivity. The file is posted onto the ipfs distributed storage network by using the infura application programming interface (API). Infura simplifies developer's IPFS network access:

Advantages:

1. Access to the IPFS file system and the Ethereum blockchain is made much more quickly available.

2. Scalable: The developers do not need to be concerned about the restrictions of the infrastructure because they manage the nodes.

3. Data Storage: Instead of storing all of the data on-chain, it is possible to store the data separately and only the hash of the data on the blockchain.

5.4 Dataset

Dataset was created before the training and recognition process. In Figure you can see the details asked from the students. Along with that 50 images of each students were collected. The reason we collected 50 images is that for the classifier we are using it is best to collected minimum 50 pictures to get the best accuracy[42]. Once the images are collected it was pre-processed and stored in 400 x 400 grayscale format in 'data folder'. The naming of the images are done by registration number of the student followed by '_' and then followed by the image count. In this out registration number will be the label on which the classification will take place.

CHAPTER 6

Implementation

6.1 Architecture

6.1.1 Registration

In Figure: 6.1 we can see that the student will come and register itself in the database in which he/she will be asked demographic details and course information. The course information was asked so that we should know that the student is getting present marked for attending the right class. Once student fills the information, it will turn on the video camera and start capturing the image. In total there will be 50 images captured. Once it is done it will pass the image to haar cascade face detector and then it will convert the image to grayscale. Post that it will same the image and details in the database.

6.1.2 Training

In Figure: 6.2 we are learning about our training process in which when a frame is is passed by the video camera it is passed to cv2's Haar Cascade Face detector which later on converted to grayscale and cropped to 400 x 400 pixels. Once it is done the images are saved in the local file. After completing the dataset and registering everyone, we are using LBPH Classifier for face recognition. After training the model on images it will return us 'clf.xml' file which again can be stored in the local files.

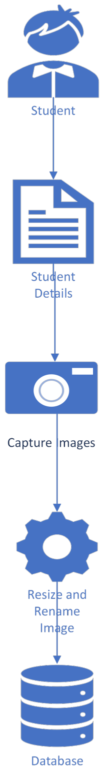


Figure 6.1: Registration Process

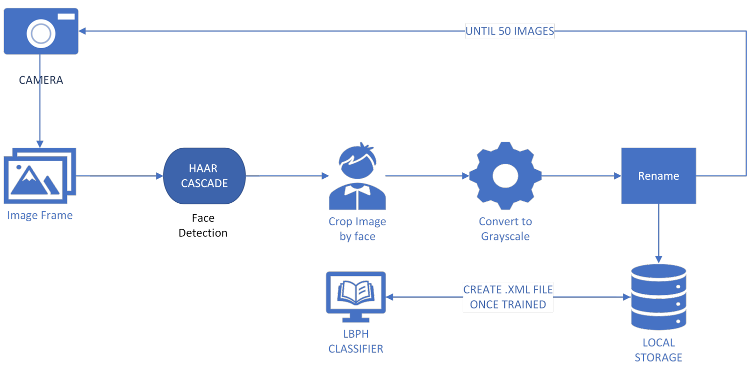


Figure 6.2: Training process

6.1.3 Recognition

We will be learning how our face recognition system is working in figure: 6.3. In real time basis our camera will pass the frames to the camera one by one. Once the frame is captures it will again follow the same pre-processing technique and see if any face is detected. Once the face is detected, it will be passed to the LBPH classifier for recognition otherwise it will go back to capture another frame. If the face is detected and passed for prediction then it will take out the confidence of the same. We have set the threshold of the confident to 75 in

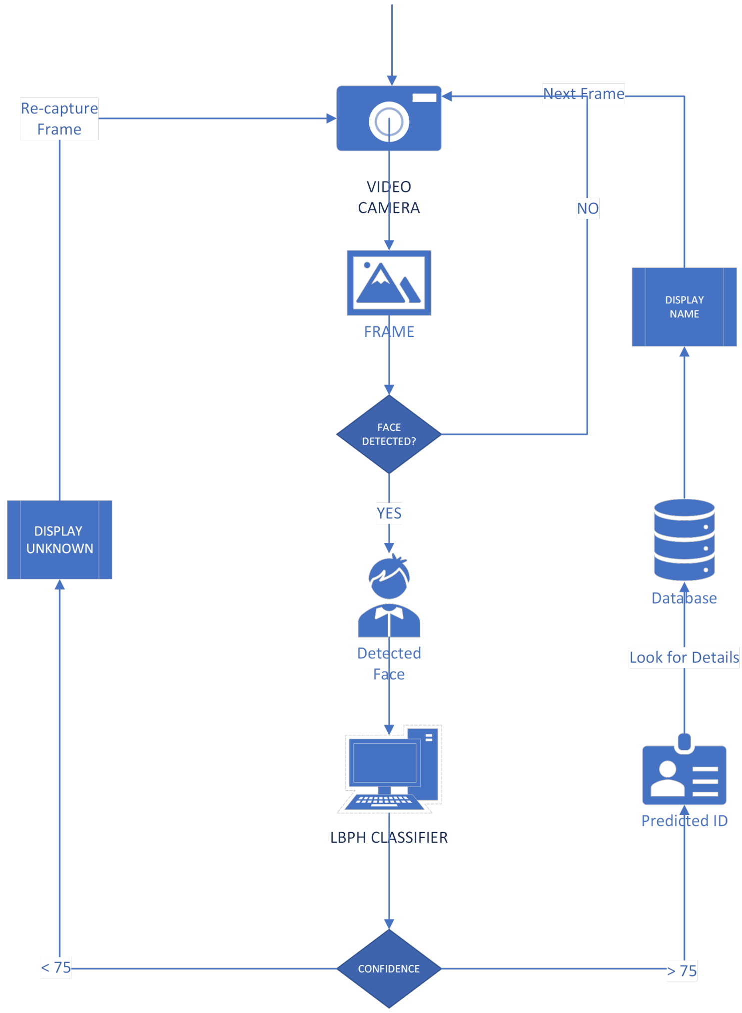


Figure 6.3: Face Recognition

which if and only if the confident is more then 75 then it will move forward otherwise it will display 'Unknown Face'. Assuming that the confidence is more than 75 which takes us to the next step of returning the label which is 'Registration ID' of a student. After getting the registration number we will pass a query in our database which will return student details to is and the attendance will be marked under the name of returned value from database.

6.1.4 Blockchain

Our facial recognition system will give us .csv file which we have to update on the blockchain. As shown in figure: 6.4, our .csv file is uploaded on an IPFS with the help of Server. Once the file is uploaded it will return us a hash value which will be regardless unique. It will return the hash value to the server which will further pass it to smart contract. In our smart contract we will be storing the unique file name and the hash value. Once the smart contract is make

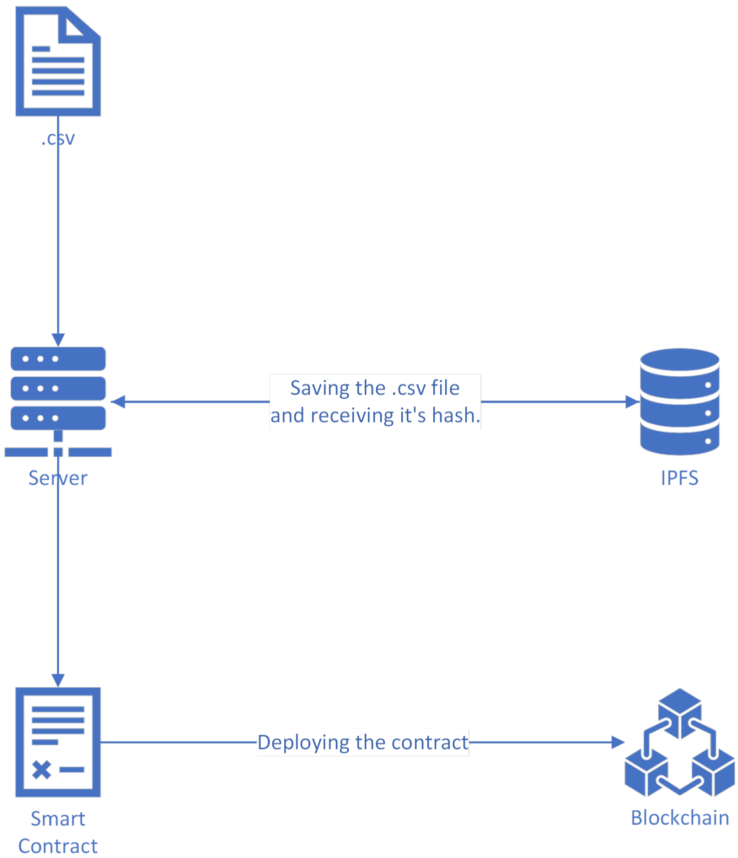


Figure 6.4: Storing .csv on Blockchain

it will be deployed on a live blockchain which makes it uneditable and transparent for all.

6.1.5 Recognition Level

In this section the software will run the python code and it will turn on your system's video camera and it will start collecting 60 frames per second(in our case). For every frame it will process the image and try to detect face in the given frame. If no face is detected then there will be no output, on the other hand if it recognise the face then it will capture the image and crop it to 400 x 400, post that it will convert the image into grayscale. Once the image is converted to grayscale it will be feeded to our classifier for recognition. Again, if the face is not classified or match any face from the database then it will show 'Unknown face' message otherwise it will fetch the ID from the local database and show the predicted student's name as shown in Figure: 6.3.

6.2 Processing Level

6.2.1 Required Modules

The modules cv2, os, image module, tkinter, mysql, and PIL are needed to carry out the facial recognition. The OpenCV module cv2 comprises the face detection and recognition algorithms. The names of the image and directory will be adjusted using the OS. Using this module, we first extract the picture names from the database directory, and then we extract registration numbers from these names, which are subsequently utilised as labels for the faces in the corresponding images. We will be using Tkinter for GUI and and mySQL will be used to connect the application with the database in which we will be storing the attendance.

6.2.2 Initialize the facial detection.

Initial step would be to identify the face out of image for which we will be using, cascade. Once we've located the area of interest in the picture that includes the face, we use it. to train the classifier. We'll employ the Haar Cascade offered by OpenCV for face detection. In the directory where OpenCV was installed, you may find the haar cascades. The face is found using the haar cascade frontal face default '.xml' algorithm. The cv2 Cascade Classifier method, which accepts the location to the cascade '.xml' file, is used to load Cascade. Relative paths are utilised if the .xml file is located in the current working directory.

6.2.3 Training

In order to develop the function that will be used to prepare the training set, we will first need to define a function that accepts the absolute path to the picture database as an argument and then returns a tuple of two lists; the first list will contain the faces that have been detected, and the second list will have the label that corresponds to each face.

6.2.4 Create the Face RecognizerObject

The following step involves the creation of the face recognizer object. OpenCV offers Eigenface Recognizer, Fisherface Recognizer, and LBPH at the moment. We performed face identification using LBPH Face Recognizer because it gives better accuracy with 50 images as mentioned in section 6.4 and it is light weight as compared to other Deep Learning models. LBP enable the description of a digital image's texture and form. This is accomplished by breaking an image into a number of small sections from which features are retrieved and utilised to determine the degree of similarity between two images.

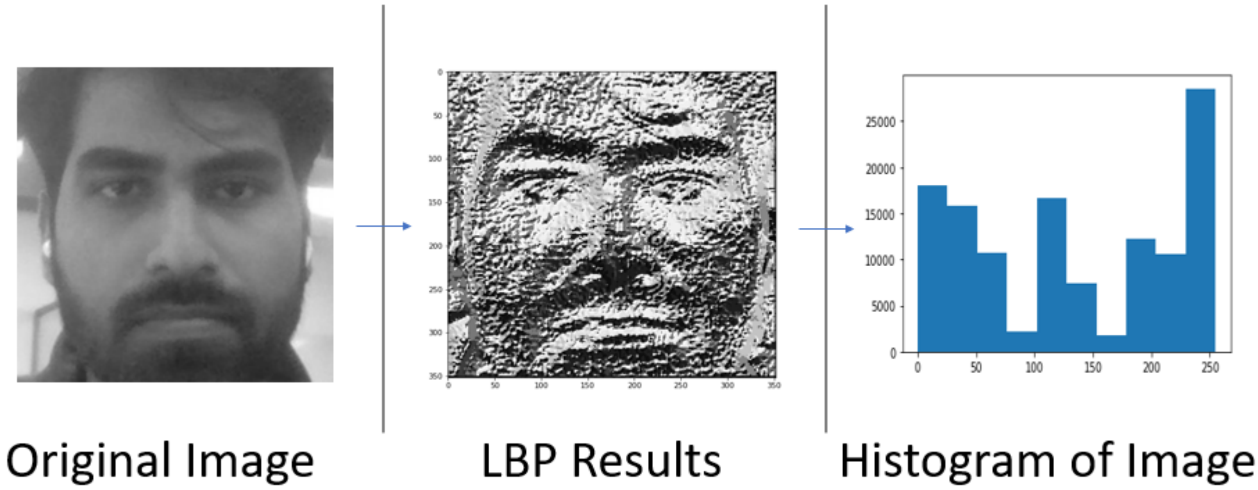


Figure 6.5: Caption

6.3 Integration with Blockchain

As a set of libraries, Web3.js facilitates communication with an Ethereum node through an HTTP or IPC link, whether that node is local or remote. In other words, it gives you JavaScript APIs to integrate with Geth(Live chain) for test network and it is written in Java. It communicates internally with geth/ganache-cli through the lightweight remote procedure call (RPC) protocol JSON-RPC.

6.3.1 Integration with Web3

Web3.js is the authorised JavaScript library for communicating with smart contracts over RPC. Since, we are using python for our application so integration with the smart contract will be easy by using it as it can communicate with any front and back end languages. Installation and operation are easy. Let's implement this integration using some basic HTML code and a Solidity-written smart contract. The code for the integration of web3 with node server was borrowed from a colleague[14].

6.3.2 Integration with IPFS

IPFS allows us to store huge files, which decreases software processing time and expense. IPFS is an efficient method for storing and exchanging huge files amongst parties that make use of blockchain technology to ensure that their data is unchangeable. Large files are broken into smaller portions. IPFS is responsible for hashing these chunks, and the hash that is generated for each chunk is completely unique because it is dependent on the data. It generates a Merkel tree that navigates to the tree's root. 5.9 illustrates how the Merkel tree appears on the IPFS network once the file has been uploaded and the hash has been created. Then, IPFS sends the smart contracts the hash of the root chunk, which is a representation of the whole file that was uploaded. After that, this hash is saved to the blockchain so that there is no need to keep enormous files or many hashes.

IPFS is utilised for the storage of the csv file of attendance. The attendance must be uploaded to IPFS, from which it may be accessed at any time and in any location. To develop the licence uploading page, we must create an IPFS client with ipfs.infura.io as the host. Using this IPFS client, the document may be uploaded to IPFS. After uploading, the user receives

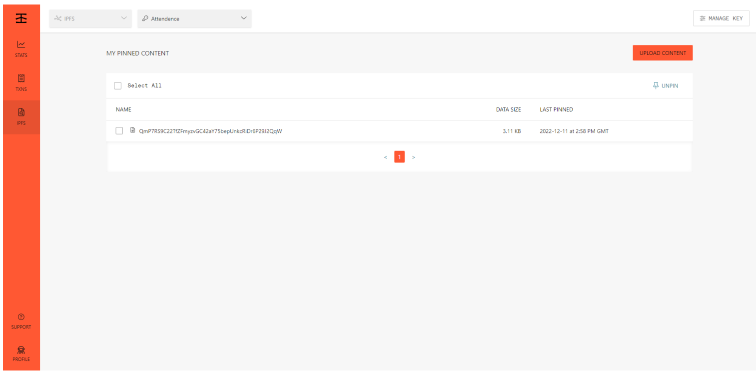


Figure 6.6: Getting Hash in return to uploading file

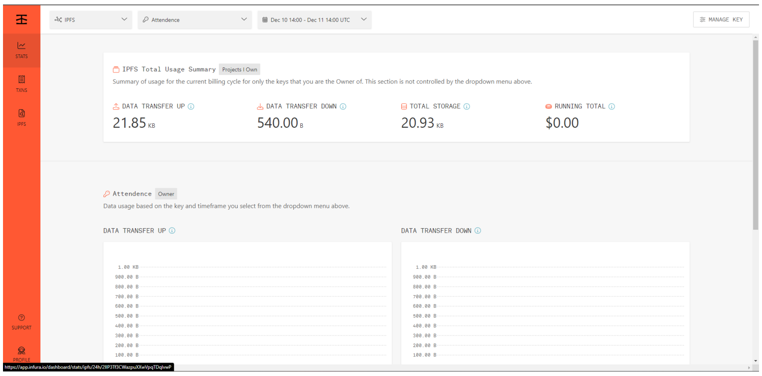


Figure 6.7: Stats of data uploading and the cost

the document's hash. Via this hash, a document posted using infura may be retrieved.

The designed user interface must be connected with the smart contract. It may be done with vanilla java script. To allow smart contract interaction with the user interface, we must load essential packages such as web3. We may build an instance of the deployed smart contract by providing its address and application programming interface (API). Using this instance, we may invoke appropriate smart contract functions.

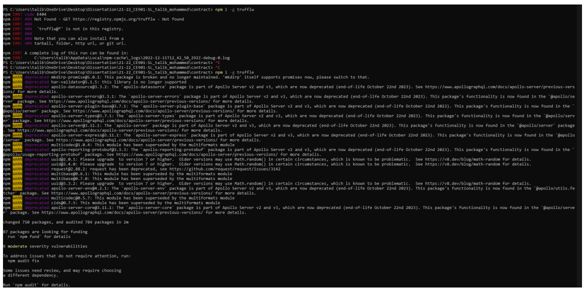
6.3.3 Truffle Suite

Truffle is a node package that requires only the execution of the node command to get up and running on Windows, Linux, or macOS.

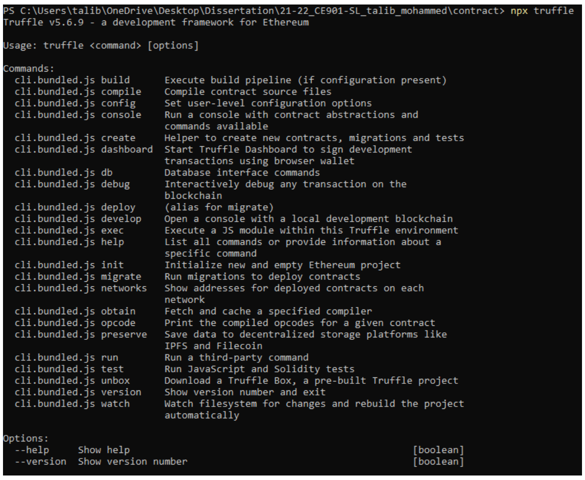
Run these 3 commands one by one: npm install -g truffle npx truffle truffle init

By using the -g option, Truffle will be installed worldwide. One must execute the ganache-cli command locally or use an Ethereum client to test a Truffle-built contract. Let's utilise ganache-cli for now.

The process of making a Truffle project is quick and easy. Start by making a new folder you may call myTruffle, then navigate to it and run the command below:

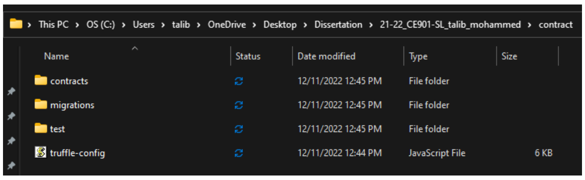


(a) Installing Truffle



(b) Creating Truffle folder

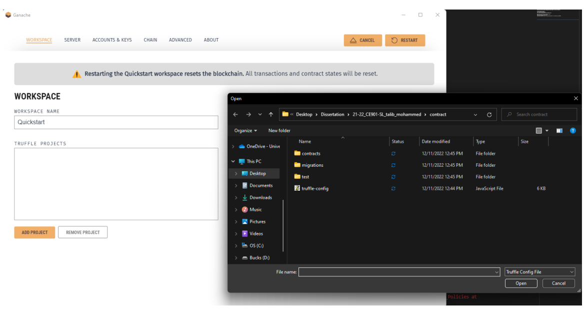
(c) Truffle init



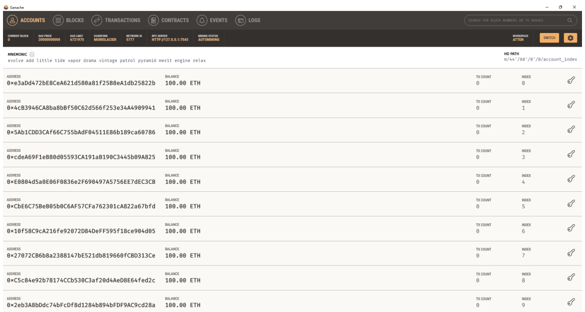
(d) Truffle project created

Figure 6.8: Running truffle init

Now open truffle suite and import the truffle-config to import the project and get the test wallets as shown in figure:6.9.



(a) Importing config file



(b) Test Wallets

Figure 6.9: Importing project

6.3.4 Smart Contract

With the help of Remix, we have written a smart contract as shown in figure: 6.10. We will be taking 2 values from the IPFS

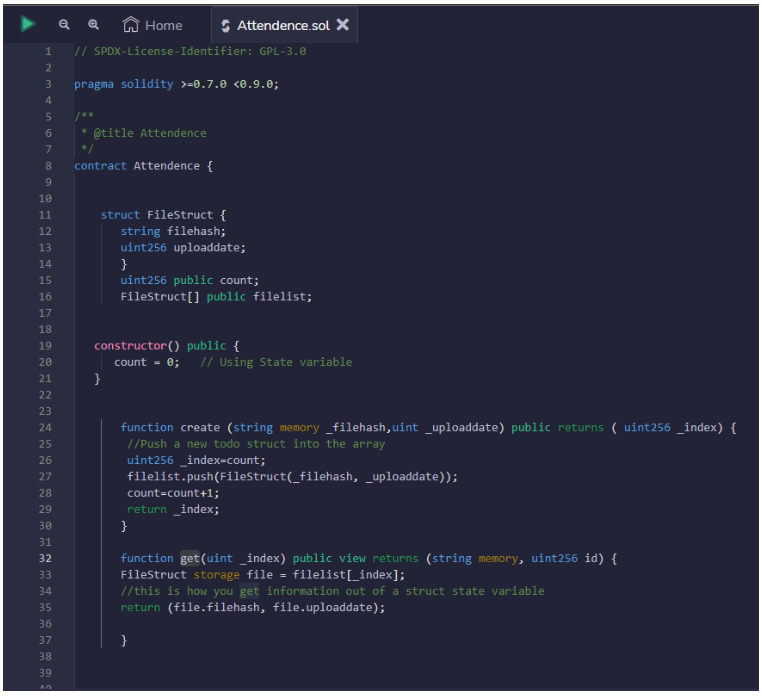


Figure 6.10: Smart Contract

6.3.5 Blockchain Deployment

Once the smart contract is compiled, we will deploy the same on local blockchain and in figure: 6.11 are the steps for the same.

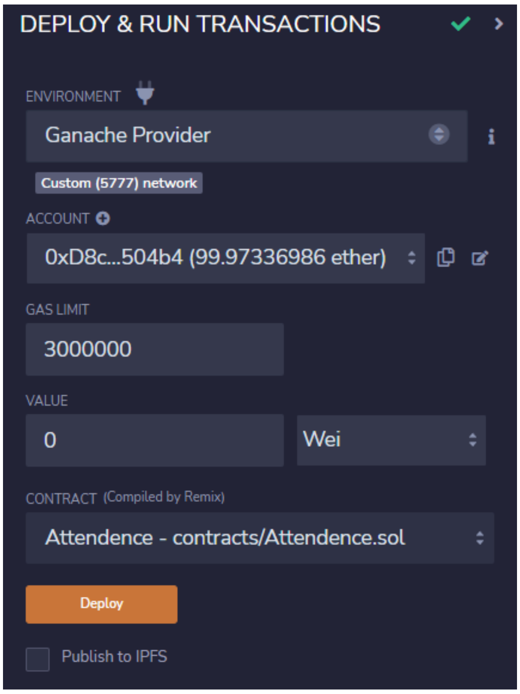
Step 1 : In step one we will go on remix where we have written and compiled smart contract. After compilation is successful we have to deploy the chain on Ganache. In figure one we will select environment as 'Ganache Provider' and Ganache JSON-RPC Endpoint as 'http://127.0.0.1:7545'.

Step 2 : Once the chain is deployed, it will pass the has received from Infura from figure:6.6, current date from node server after converting into unix epoch time and then click on transact to complete the transaction.

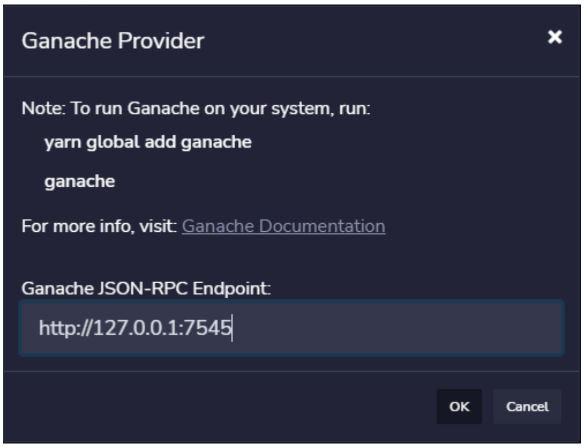
6.4 User Interface

6.4.1 Admin's Registration

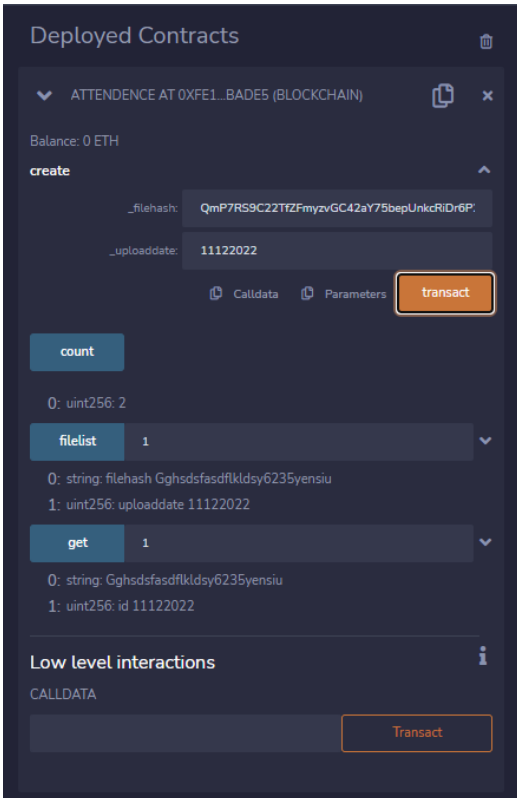
Figure: 6.12 shows how the registration page looks like and how the data is stored in data base and once the data is stored, it will be used for the login credentials.



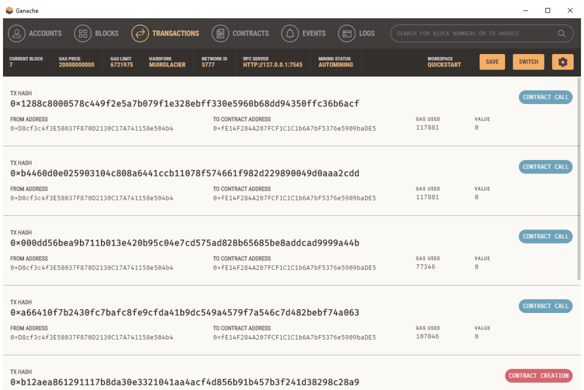
(a) Installing Truffle



(b) Creating Truffle folder

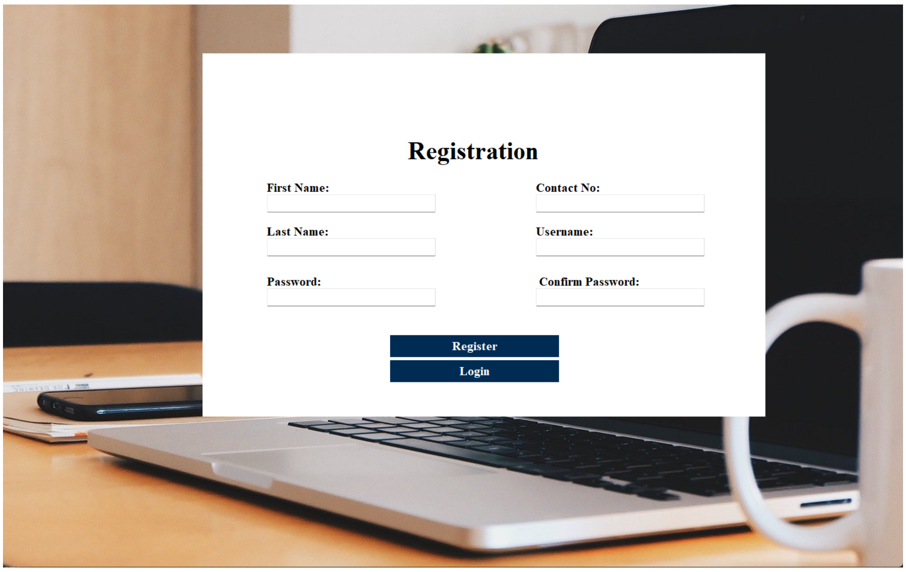


(c) Truffle init



(d) Truffle project created

Figure 6.11: Deployment



(a) Registration page

registration	prid	name	department	time	date	attendance
112345	TALIB92708	Mohammed Talib	Computer Science	20:35:54	25/11/2022	Present
NULL	NULL	NULL	NULL	NULL	NULL	NULL

(b) Data Base - Admins

Figure 6.12: Registration for Admin

6.4.2 Login Page

Figure: 6.13 is the login page which only accepts admin's information and the credentials will be validated from 6.12b.

6.4.3 Dashboard

Figure: 6.14 shows the dashboard in which we get 4 options.

- Student Details
- Training
- Face Recognition
- Attendance

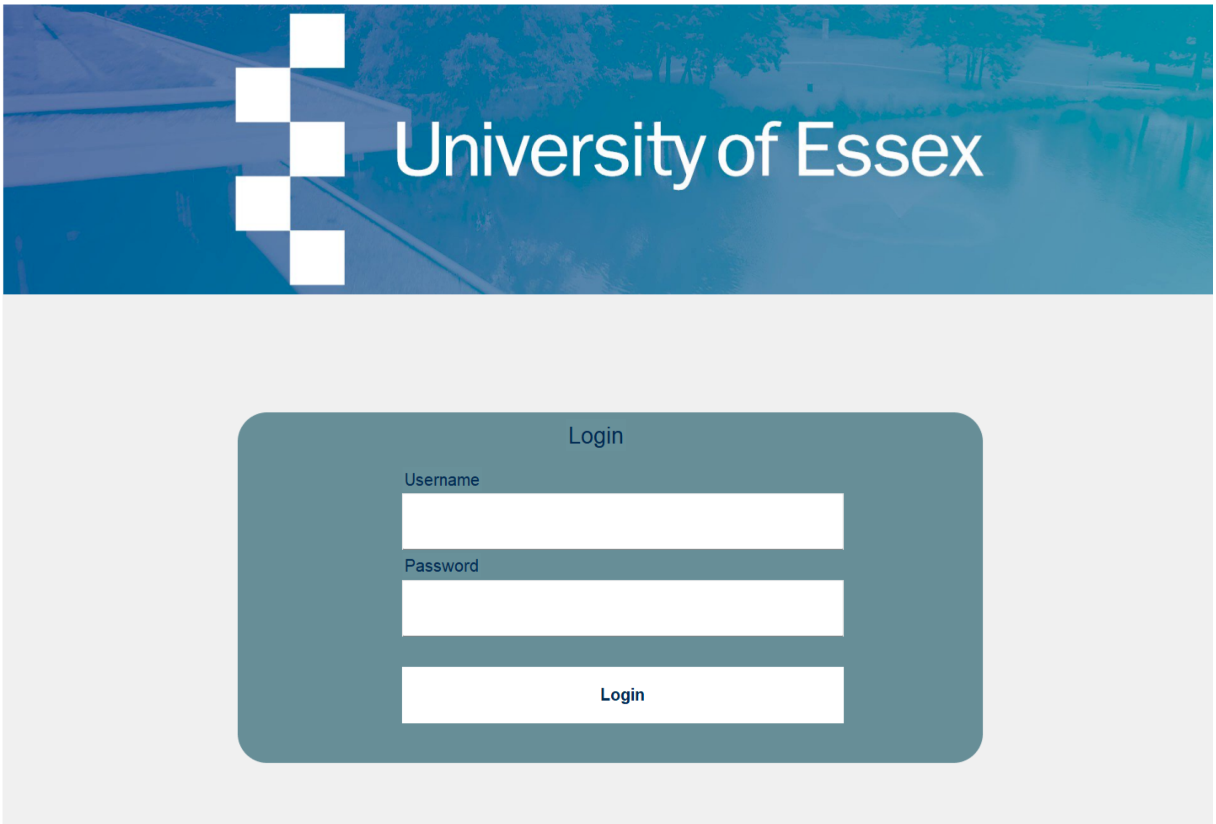


Figure 6.13: Login Page

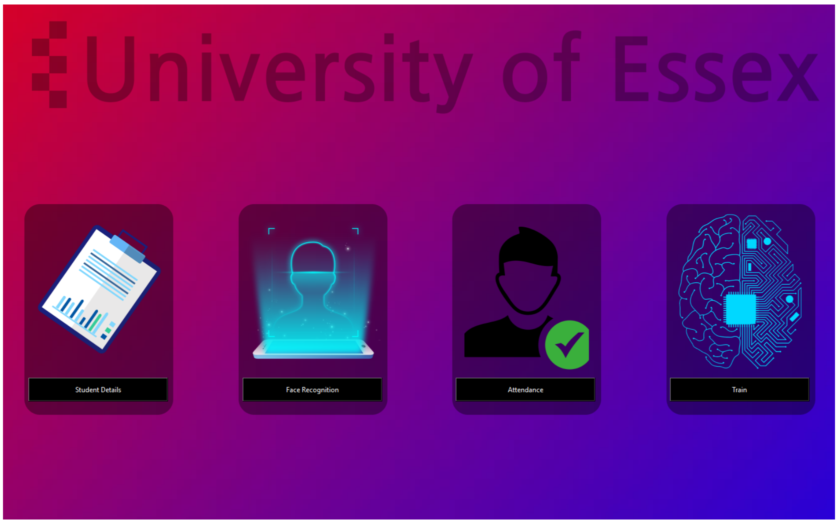
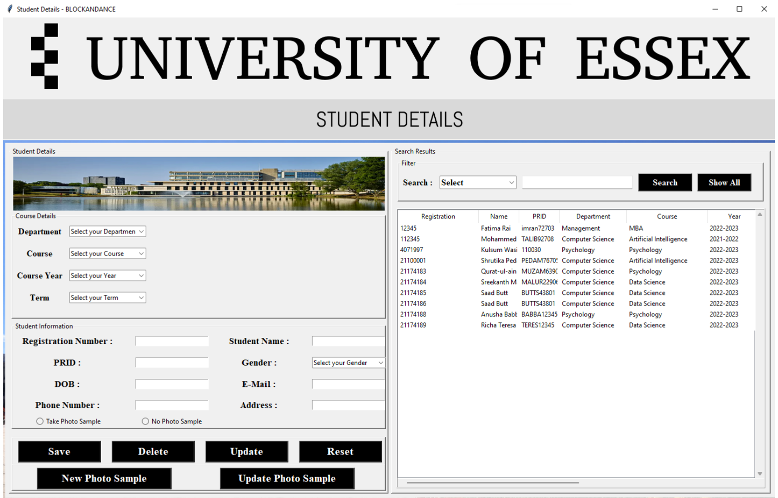


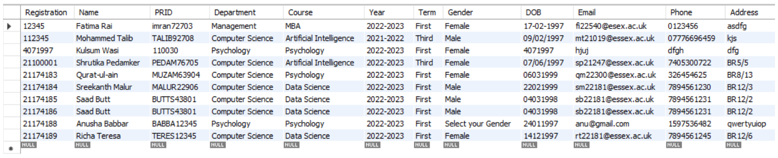
Figure 6.14: Dashboard

6.4.4 Student's Registration

Figure: 6.15 is admin accessed registration page in which they can help other students to register into the database. Figure: 6.17a shows the registration page and Figure: 6.17b shows the database in which the data is being stored.



(a) Registration page



Registration	Name	PRID	Department	Course	Year	Term	Gender	DOB	Email	Phone	Address
12345	Fatima Rai	imran72703	Management	MBA	2022-2023	First	Female	17-02-1997	fi22540@esex.ac.uk	0123456	asdfg
112345	Mohammed Talib	TALIB92708	Computer Science	Artificial Intelligence	2021-2022	Third	Male	09/02/1997	mt21019@essex.ac.uk	07776696459	kjs
4071997	Kulsum Wasi	110030	Psychology	Psychology	2022-2023	First	Female	4071997	hjuj	dfgh	dfg
21100001	Shrutika Pedamker	PEDAM76705	Computer Science	Artificial Intelligence	2022-2023	Third	Female	07/06/1997	sp21247@essex.ac.uk	7405300722	BR5/5
21174183	Qurat-ul-ain	MUZAM63904	Computer Science	Psychology	2022-2023	First	Female	06031999	qm22300@essex.ac.uk	326454625	BR8/13
21174184	Sreekanth Malur	MALUR22906	Computer Science	Data Science	2022-2023	First	Male	22021999	sm22181@essex.ac.uk	7894561230	BR12/3
21174185	Saad Butt	BUTTS43801	Computer Science	Data Science	2022-2023	First	Male	04031998	sb22181@essex.ac.uk	7894561231	BR12/2
21174186	Saad Butt	BUTTS43801	Computer Science	Data Science	2022-2023	First	Male	04031998	sb22181@essex.ac.uk	7894561231	BR12/2
21174188	Anusha Babbar	BABBA12345	Psychology	Psychology	2022-2023	First	Select your Gender	24011997	anu@gmail.com	1597536482	qwertyuiop
21174189	Richa Teresa	TERES12345	Computer Science	Data Science	2022-2023	First	Female	14121997	rt22181@essex.ac.uk	7894561245	BR12/6
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

(b) Data Base - Student Details

Figure 6.15: Registration page for Students

6.4.5 Training

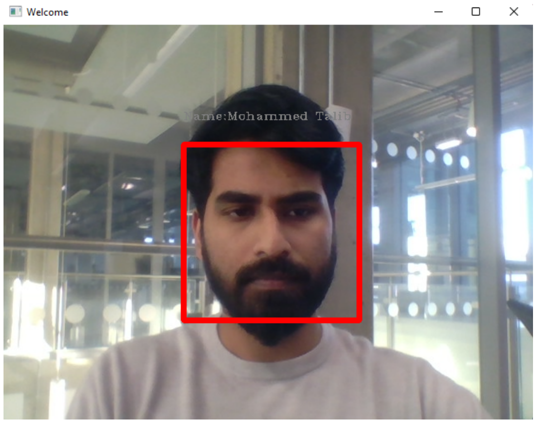
This section will start training the data as explained in Section: 6.1.2.

Figure 6.16: Training

6.4.6 Face Recognition

Figure: 6.17 is the face recognition which will open the web camera and start prediction as explained in section: 6.1.3.

(a) Recognition page

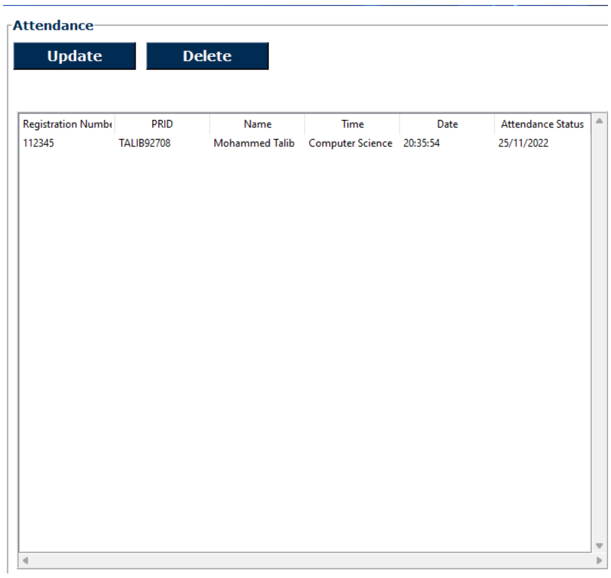


(b) Detection on a real time basis

Figure 6.17: Facial Recognition

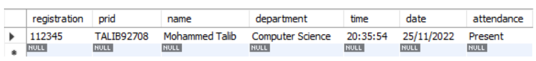
6.4.7 Attendance

Figure: 6.13 is the login page which only accepts admin's information.



Registration Numbe	PRID	Name	Time	Date	Attendance Status
112345	TALIB92708	Mohammed Talib	Computer Science	20:35:54	25/11/2022

(a) Attendance on the App



registration	prid	name	department	time	date	attendance
112345	TALIB92708	Mohammed Talib	Computer Science	20:35:54	25/11/2022	Present

(b) Database - Attendance

Figure 6.18: Marked Attendance

CHAPTER 7

Evaluation and Results

7.1 Face Recognition

7.1.1 Evaluation Technique

While evaluating the performance, we should not focus on accuracy itself. Accuracy may help you to reach the maximum number of correct guesses; however, we must focus on the wrong evaluation as well as it can be fatal. For evaluation we have several techniques, such as: accuracy, recall, precision, and f1 score. We can also evaluate the performance by confusion matrix. Let's discuss these term one by one:

Confusion Matrix: It is a study of prediction of a trained model. All the correct and incorrect prediction are summed up in a table and helps us the study accuracy, precision and recall. It mainly consists of 4 categories:

- True Positive (TP): All the labels that were correctly predicted as Positive.
- True Negative (TN): All the labels that were correctly predicted as Negative.
- False Positive (FP): Labels that were Negative but predicted as Positive also known as Type 1 error.
- False Negative (FN): Labels that were Positive but predicted as Negative also known as Type 2 error and it can be fatal.

Now lets understand the classification report:

- Accuracy: Accuracy is referred to the number of correct guesses and it can be calculates by:

  $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

- Recall: It is the ability of a machine learning algorithm to predict all positive instances. It can be calculated by the ration of TP to the sum of TP and FN.

  $Recall = \frac{TP}{TP+FN}$

- F1-Score: It is a weighted harmonic mean of precision and recall. We should always consider F1 score of a model to compare the performance in place of accuracy.

  $F1 = \frac{2*Precision*Recall}{Precision+Recall} = \frac{2*TP}{2*TP+FP+FN}$

- Precision: It is the ability of a machine learning algorithm not to predict negative instance as positive. It can be calculated by the ration of TP to the sum of TP and FP.

  $Precision = \frac{TP}{TP+FP}$

- Support: It is the occurrence of a class in a particular data. Adding up all the rows of confusion matrix can give us the support.

7.1.2 Results

We have taken 4 scenarios in which we have played around with 2 cameras and lighting quality. All the 4 scenarios are as below:

- Scenario1: 4K camera with low lighting.
- Scenario2: 4K camera with good lighting.
- Scenario3: 720P HD camera with good lighting.
- Scenario4: 720P HD camera with low lighting.

We have seen that with 4K cameras we did not receive any different results so we will be displaying them exactly the same.

Accuracy

We have noticed in the figure: 7.1 for 4K camera with low and good lighting we have received the same accuracy and f-1 score which is 100%. For 720P HD camera with good lighting the f-1 score was 99.7% anf the accuracy was %99.8. Where as in 720P HD camera with low lighting the accuracy and f-1 score went down to 95.1% and 95.0% respectively.

```
f1_score(y_test,predictions,average="macro")
1.0

[48] accuracy_score(y_test,predictions)
1.0
```

Scenario 1

```
f1_score(y_test,predictions,average="macro")
0.997999199679872

[42] accuracy_score(y_test,predictions)
0.998
```

Scenario 3

```
[61] f1_score(y_test,predictions,average="macro")
0.9509719133968856

[62] accuracy_score(y_test,predictions)
0.951063829787234
```

Scenario 4

Figure 7.1: Accuracy and F-1 Score

Classification Report

Below in figure7.2 are the classification reports of all the 3 scenarios and we found out that the best results received with good lighting and 4K camera

	precision	recall	f1-score	support
1	1.00	1.00	1.00	20
2	1.00	1.00	1.00	20
3	1.00	1.00	1.00	20
4	1.00	1.00	1.00	20
5	1.00	1.00	1.00	20
6	1.00	1.00	1.00	20
7	1.00	1.00	1.00	20
10	1.00	1.00	1.00	20
11	1.00	1.00	1.00	20
123	1.00	1.00	1.00	20
12345	1.00	1.00	1.00	20
4071997	1.00	1.00	1.00	20
21100001	1.00	1.00	1.00	20
21174183	1.00	1.00	1.00	20
21174184	1.00	1.00	1.00	20
21174186	1.00	1.00	1.00	20
21174188	1.00	1.00	1.00	20
21174189	1.00	1.00	1.00	20
21174190	1.00	1.00	1.00	20
21175191	1.00	1.00	1.00	20
accuracy			1.00	400

Scenario 1

	precision	recall	f1-score	support
1	1.00	1.00	1.00	25
2	1.00	1.00	1.00	25
3	1.00	1.00	1.00	25
4	1.00	1.00	1.00	25
5	1.00	0.96	0.98	25
6	1.00	1.00	1.00	25
7	1.00	1.00	1.00	25
10	1.00	1.00	1.00	25
11	1.00	1.00	1.00	25
123	1.00	1.00	1.00	25
12345	1.00	1.00	1.00	25
4071997	1.00	1.00	1.00	25
21100001	1.00	1.00	1.00	25
21174183	1.00	1.00	1.00	25
21174184	1.00	1.00	1.00	25
21174186	1.00	1.00	1.00	25
21174188	1.00	1.00	1.00	25
21174189	0.96	1.00	0.98	25
21174190	1.00	1.00	1.00	25
21175191	1.00	1.00	1.00	25
accuracy			1.00	500

Scenario 3

	precision	recall	f1-score	support
1	0.96	1.00	0.98	47
2	0.98	1.00	0.99	47
3	0.60	0.77	0.67	47
4	0.98	1.00	0.99	47
5	1.00	0.96	0.98	47
6	1.00	1.00	1.00	47
7	1.00	1.00	1.00	47
10	1.00	0.96	0.98	47
11	1.00	1.00	1.00	47
123	1.00	1.00	1.00	47
12345	1.00	1.00	1.00	47
4071997	1.00	1.00	1.00	47
21100001	1.00	1.00	1.00	47
21174183	1.00	1.00	1.00	47
21174184	1.00	1.00	1.00	47
21174186	0.98	0.96	0.97	47
21174188	1.00	1.00	1.00	47
21174189	0.96	1.00	0.98	47
21174190	0.61	0.49	0.54	47
21175191	1.00	0.89	0.94	47
accuracy			0.95	940

Scenario 4

Figure 7.2: Classification Report

Confusion Matrix

Below in figure7.3 are the confusiion matrix' of all the 3 scenarios and we found out that the best results received with good lighting and 4K camera

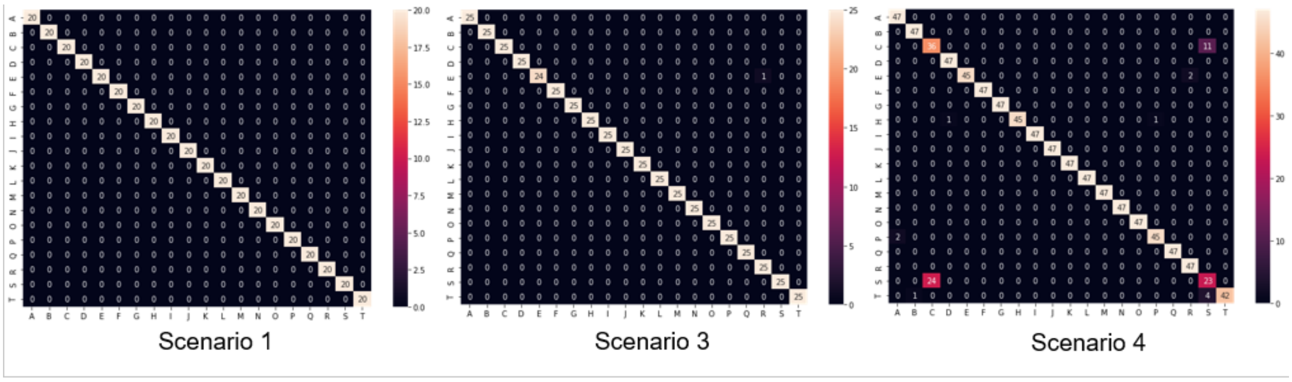


Figure 7.3: Confusion Matrix

7.2 Database

To evaluate the database, we will have to check if the data is being stored in the correct manner as passing through the app. Figure 6.17b, 6.18b, and 6.12b shows the result of passing the data and getting correctly updated which denotes that the our application is connected with the database correctly and below 7.4is the screenshot of the connection established as well.

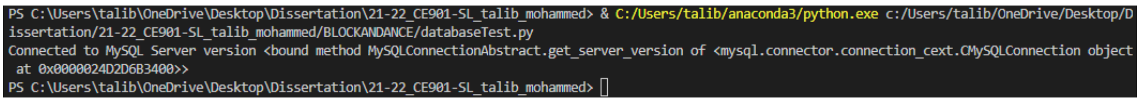


Figure 7.4: Database Test

7.3 Blockchain

Once a transaction was made, our blockchain will return the block address of the same. With the help of block address we will be able to get the hash of the following transaction and with that hash we will be able to access the file belongs to that transaction. Results of the following are provided in Figure 6.5 and Figure 6.10. It shows that we are getting hash and are being able to create a block of the transaction with the same. We were able to execute the smart contract successfully and it is returning the block value of the same

7.4 Blockendance

We evaluated the application by taking the feedback from the subjects that how user friendly was the application for them. Below are the positive and negative points extracted from their experience and feedback.

- Registration of Admin and Students are simple.
- Accuracy of the real-time face recognition was almost correct everytime.
- There are a few question that were asked by the application were irrelevant.
- User Interface of the application was basic and which can be looked better.

CHAPTER 8

Conclusion

We attempted to propose a real-time based facial recognition system in this post for usage as university attendance and authentication systems. Blockandacnce is an agile, secure, user-friendly, cost- and time-effective technology. With the aid of cutting-edge ideas like facial recognition, this software can increase the efficiency and security of colleges' boring everyday attendance taking procedures. With the help of Haar Cascade we will be able to detect the faces in real time basis and after LBPH Recognizer helped us to recognise the face and mark the attendance. Since, HAAR and LBPH as light weight models so it took less computational power we did not see any major lag in real-time predictions. Even in the evaluation it we received the accuracy above 95%.

Using the camera system's capabilities, a student's attendance at a lecture may be documented in a register. To construct such a system and provide safe data storage, it is practical to employ a blockchain. Regrettably, the scientific publications associated with blockchain technology are just beginning to become more extensively disseminated; hence, there are relatively few scientific literature describing the blockchains' operation and structure.

By implementing a private blockchain-based attendance system, university will have full administrative powers and may manage blockchain access in accordance with its regulations.

To completely automate the system, it is advised to employ smart contracts. Due to the fact that the contract regulates data transit, care must be taken to ensure that it is error-free. The identification and transmission and reception of data occur in accordance with the terms of the contract. The smart contract performs the conditions it contains automatically.

CHAPTER 9

Future Work

Since, the study has scope of improvement so there are a few points we would like to work on further:

- We noticed that input images plays an important role in face recognition and detection so we would like to work on the pre-processing of the images so that the accuracy can be increased even in the low lighting and we will test try to optimize LBPH classifier even more for the optimum solution.
- Security and integrity of blockchain were not able to evaluate, so in future we will search for a general method to test the blockchain.
- Integration of blockchain and attendance management system is what we will be focusing on next.
- UI of the application can be improved to make it bit mordern

APPENDIX A

Gitlab Submission

The code for the dissertation can be found at Gitlab Repository Link.

Bibliography

[1] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face recognition with local binary patterns. In *European conference on computer vision*, pages 469–481. Springer, 2004.

[2] Tareq Ahram, Arman Sargolzaei, Saman Sargolzaei, Jeff Daniels, and Ben Amaba. Blockchain technology innovations. In *2017 IEEE technology & engineering management conference (TEMSCON)*, pages 137–141. IEEE, 2017.

[3] Khoula Al Harthy, Fatma Al Shuhaimi, and Khalid Khalifa Juma Al Ismaily. The upcoming blockchain adoption in higher-education: requirements and process. In *2019 4th MEC international conference on big data and smart city (ICBDSC)*, pages 1–5. IEEE, 2019.

[4] Grigore Albeanu. Blockchain technology and education. In *The 12th International Conference on Virtual Learning ICVL*, pages 271–275, 2017.

[5] Muhammad Salek Ali, Massimo Vecchio, Miguel Pincheira, Koustabh Dolui, Fabio Antonelli, and Mubashir Husain Rehmani. Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 21(2):1676–1717, 2018.

[6] Nusrat Mubin Ara, Nishikanto Sarkar Simul, and Md Saiful Islam. Convolutional neural network approach for vision based student recognition system. In *2017 20th International Conference of Computer and Information Technology (ICCIT)*, pages 1–6. IEEE, 2017.

[7] Marko Arsenovic, Srdjan Sladojevic, Andras Anderla, and Darko Stefanovic. Facetimeâdeep learning based face recognition attendance system. In *2017 IEEE 15th International symposium on intelligent systems and informatics (SISY)*, pages 000053–000058. IEEE, 2017.

[8] Mirko Avantaggiato and Pierluigi Gallo. Challenges and opportunities using multi-chain for real estate. In *2019 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pages 1–5. IEEE, 2019.

[9] Naveed Khan Balcoh, M Haroon Yousaf, Waqar Ahmad, and M Iram Baig. Algorithm for efficient attendance management: Face recognition based approach. *International Journal of Computer Science Issues (IJCSI)*, 9(4):146, 2012.

[10] Marian Stewart Bartlett, Javier R Movellan, and Terrence J Sejnowski. Face recognition by independent component analysis. *IEEE Transactions on neural networks*, 13(6):1450–1464, 2002.

[11] Shubhobrata Bhattacharya, Gowtham Sandeep Nainala, Prosenjit Das, and Aurobinda Routray. Smart attendance monitoring system (sams): a face recognition based attendance system for classroom environment. In *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)*, pages 358–360. IEEE, 2018.

[12] Wei Cai, Zehua Wang, Jason B Ernst, Zhen Hong, Chen Feng, and Victor CM Leung. Decentralized applications: The blockchain-empowered software system. *IEEE Access*, 6:53019–53033, 2018.

[13] Junyi Chai, Hao Zeng, Anming Li, and Eric W.T. Ngai. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications*, 6:100134, 2021.

[14] Umer Chaudhary. https://github.com/muhammadumerriaz.

[15] Mohammad Jabed Morshed Chowdhury, Alan Colman, Muhammad Ashad Kabir, Jun Han, and Paul Sarda. Blockchain versus database: A critical analysis. In *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pages 1348–1353, 2018.

[16] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 886–893. Ieee, 2005.

[17] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019.

[18] Oscar Déniz, Gloria Bueno, Jesús Salido, and Fernando De la Torre. Face recognition using histograms of oriented gradients. *Pattern recognition letters*, 32(12):1598–1603, 2011.

[19] Stephen Devadoss and John Foltz. Evaluation of factors influencing student class attendance and performance. *American Journal of Agricultural Economics*, 78(3):499–507, 1996.

[20] Massimo Di Pierro. What is the blockchain? *Computing in Science & Engineering*, 19(5):92–95, 2017.

[21] Lahiru Dinalankara. Face detection & face recognition using open computer vision classifies. *ResearchGate*, 2017.

[22] Shalince Dominic, C Aparna, Aswin S Nath, Mahesh Mohan, MS Ajeesh, and Anil Antony. A review of face detection system. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 3536–3539. IEEE, 2016.

[23] Ji-Xiang Du, De-Shuang Huang, Guo-Jun Zhang, and Zeng-Fu Wang. A novel full structure optimization algorithm for radial basis probabilistic neural networks. *Neurocomputing*, 70(1-3):592–596, 2006.

[24] Sony Global Education. Sony global education develops technology using blockchain for open sharing of academic proficiency and progress records [press release], 2016.

[25] Hussam Elbehiery. Enhancement of qr code studentâs attendance management system using gps. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 21(4):18–30, 2019.

[26] S Georgopoulos. Sony uses blockchain technology to digitize education records. *Neowin*, 2017.

[27] Kruti Goyal, Kartikey Agarwal, and Rishi Kumar. Face detection and tracking: Using opencv. In *2017 International conference of Electronics, Communication and Aerospace Technology (ICECA)*, volume 1, pages 474–478. IEEE, 2017.

[28] Steven E Gump. The cost of cutting class: Attendance as a predictor of success. *College Teaching*, 53(1):21–26, 2005.

[29] Fei Han and De-Shuang Huang. A new constrained learning algorithm for function approximation by encoding a priori information into feedforward neural networks. *Neural Computing and Applications*, 17(5):433–439, 2008.

[30] Fei Han, Qing-Hua Ling, and De-Shuang Huang. Modified constrained learning algorithms incorporating additional functional constraints into neural networks. *Information Sciences*, 178(3):907–919, 2008.

[31] Fei Han, Qing-Hua Ling, and De-Shuang Huang. An improved approximation approach incorporating particle swarm optimization and a priori information into neural networks. *Neural Computing and Applications*, 19(2):255–261, 2010.

[32] WK Hon, John Palfreyman, and Matthew Tegart. Distributed ledger technology & cybersecurity. *European Union Agency For Network And Information Securit (ENISA)*, 2016.

[33] De-Shuang Huang and Ji-Xiang Du. A constructive hybrid structure optimization methodology for radial basis probabilistic neural networks. *IEEE Transactions on neural networks*, 19(12):2099–2115, 2008.

[34] De-Shuang Huang, Horace Ho-Shing Ip, Ken Chee Keung Law, and Zheru Chi. Zeroing polynomials using modified constrained neural network approach. *IEEE Transactions on Neural Networks*, 16(3):721–732, 2005.

[35] De-Shuang Huang and Wen Jiang. A general cpl-ads methodology for fixing dynamic parameters in dual environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(5):1489–1500, 2012.

[36] De-Shuang Huang and Song-De Ma. Linear and nonlinear feedforward neural network classifiers: a comprehensive understanding. *Journal of Intelligent Systems*, 9(1):1–38, 1999.

[37] De-Shuang Huang and Jian-Xun Mi. A new constrained independent component analysis method. *IEEE Transactions on Neural networks*, 18(5):1532–1535, 2007.

[38] De-Shuang Huang and Wen-Bo Zhao. Determining the centers of radial basis probabilistic neural networks by recursive orthogonal least square algorithms. *Applied Mathematics and Computation*, 162(1):461–473, 2005.

[39] De-Shuang Huang, Xing-Ming Zhao, Guang-Bin Huang, and Yiu-Ming Cheung. Classifying protein sequences using hydropathy blocks. *Pattern recognition*, 39(12):2293–2300, 2006.

[40] DS Huang. The study of data mining methods for gene expression profiles, 2009.

[41] Marco Iansiti and Karim Lakhani. The truth about blockchain:. *Harvard business review*, 95:118–127, 01 2017.

[42] A. M. Jagtap, Vrushabh Kangale, Kushal Unune, and Prathmesh Gosavi. A study of lbph, eigenface, fisherface and haar-like features for face recognition using opencv. In *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, pages 219–224, 2019.

[43] Vidit Jain and Erik Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical report, UMass Amherst technical report, 2010.

[44] Umarani Jayaraman, Phalguni Gupta, Sandesh Gupta, Geetika Arora, and Kamlesh Tiwari. Recent development in face recognition. *Neurocomputing*, 408:231–245, 2020.

[45] Abhishek Jha. Class room attendance system using facial recognition system. *The International Journal of Mathematics, Science, Technology and Management*, 2(3):4–7, 2007.

[46] Kushsairy Kadir, Mohd Khairi Kamaruddin, Haidawati Nasir, Sairul I Safie, and Zulkifli Abdul Kadir Bakti. A comparative study between lbp and haar-like features for face detection using opencv. In *2014 4th International Conference on Engineering Technology and Technopreneuship (ICE2T)*, pages 335–339. IEEE, 2014.

[47] Maddu Kamaraju and Penta Anil Kumar. Wireless fingerprint attendance management system. In *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–6. IEEE, 2015.

[48] Yohei Kawaguchi, Tetsuo Shoji, Weijane Lin, Koh Kakusho, and Michihiko Minoh. Face recognition-based lecture attendance system. In *The 3rd AEARU workshop on network education*, pages 70–75, 2005.

[49] Kwang In Kim, Keechul Jung, and Hang Joon Kim. Face recognition using kernel principal component analysis. *IEEE signal processing letters*, 9(2):40–42, 2002.

[50] Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009.

[51] Nir Kshetri. Blockchain's roles in strengthening cybersecurity and protecting privacy. *Telecommunications policy*, 41(10):1027–1038, 2017.

[52] Karim R Lakhani and M Iansiti. The truth about blockchain. *Harvard Business Review*, 95(1):119–127, 2017.

[53] Shengcai Liao, Anil K Jain, and Stan Z Li. A fast and accurate unconstrained face detector. *IEEE transactions on pattern analysis and machine intelligence*, 38(2):211–223, 2015.

[54] Rainer Lienhart and Jochen Maydt. An extended set of haar-like features for rapid object detection. In *Proceedings. international conference on image processing*, volume 1, pages I–I. IEEE, 2002.

[55] Jia-Jing Lin and Shih-Chang Huang. The implementation of the visitor access control system for the senior citizen based on the lbp face recognition. In *2017 International Conference on Fuzzy Theory and Its Applications (iFUZZY)*, pages 1–6. IEEE, 2017.

[56] Shang-Hung Lin, Sun-Yuan Kung, and Long-Ji Lin. Face recognition/detection by probabilistic decision-based neural network. *IEEE transactions on neural networks*, 8(1):114–132, 1997.

[57] Kun-Hong Liu and De-Shuang Huang. Cancer classification using rotation forest. *Computers in biology and medicine*, 38(5):601–610, 2008.

[58] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017.

[59] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[60] Ze Lu, Xudong Jiang, and Alex Kot. Deep coupled resnet for low-resolution face recognition. *IEEE Signal Processing Letters*, 25(4):526–530, 2018.

[61] Iacopo Masi, Feng-Ju Chang, Jongmoo Choi, Shai Harel, Jungyeon Kim, Kanggeon Kim, Jatuporn Leksut, Stephen Rawls, Yue Wu, Tal Hassner, et al. Learning pose-aware models for pose-invariant face recognition in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):379–393, 2018.

[62] Iacopo Masi, Yue Wu, Tal Hassner, and Prem Natarajan. Deep face recognition: A survey. In *2018 31st SIBGRAPI conference on graphics, patterns and images (SIBGRAPI)*, pages 471–478. IEEE, 2018.

[63] M Meyliana, Cadelina Cassandra, Henry Antonius Eka Surjandy, Erick Fernando Widjaja, Harjanto Prabowo, and Charles Joseph. Defying the certification diploma forgery with blockchain platform: a proposed model. In *Proceedings of the International Conferences ICT, Society, and Human Beings 2019; Connected Smart Cities 2019; and Web Based Communities and Social Media 2019*, pages 63–71, 2019.

[64] Yakob Utama Chandra Meyliana, Surjandy Cadelina Cassandra, and Henry Antonius Eka Widjaja Erick Fernando. Recording of student attendance with blockchain technology to avoid fake presence data in teaching learning process.

[65] Rachel Muncrief and Ben Sharpe. Overview of the heavy-duty vehicle market and co2 emissions in the european union. *Washington, DC: The International Council on Clean Transportation*, 2015.

[66] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.

[67] Lucas M Palma, Martín AG Vigil, Fernando L Pereira, and Jean E Martina. Blockchain and smart contracts for higher education registry in brazil. *International Journal of Network Management*, 29(3):e2061, 2019.

[68] Dimitri Pissarenko. Eigenface-based facial recognition. *December 1st*, 2002.

[69] Adri Priadana and Muhammad Habibi. Face detection using haar cascades to filter selfie face image on instagram. In *2019 International Conference of Artificial Intelligence and Information Technology (ICAIIT)*, pages 6–9. IEEE, 2019.

[70] AA Abd Rahni, N Zainal, MF Zainal Adna, NE Othman, and MF Bukhori. Development of the online student attendance monitoring system (samstm) based on qr-codes and mobile devices. *J. Eng. Sci. Technol*, 10:28–40, 2015.

[71] Henry A Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *IEEE Transactions on pattern analysis and machine intelligence*, 20(1):23–38, 1998.

[72] MA SafiIe, R Hartono, and G Pratama. The development of student attandance system using rfid and internet of things (iot) technology. In *IOP Conference Series: Materials Science and Engineering*, volume 578, page 012084. IOP Publishing, 2019.

[73] Kelvin Salton. Face recognition: Understanding lbph algorithm. *Towards Data Science*, 10, 2017.

[74] Rüdiger Schollmeier. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In *Proceedings First International Conference on Peer-to-Peer Computing*, pages 101–102. IEEE, 2001.

[75] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.

[76] Mike Sharples and John Domingue. *Chapter:'The Blockchain and Kudos: A Distributed System for Educational Record, Reputation and Reward'from book: Adaptive and Adaptable Learning: 11th European Conference on Technology Enhanced Learning, EC-TEL 2016, Lyon, France, September 13-16, 2016, Proceedings*. Springer Nature, 2016.

[77] Manjot Singh, Md Alam Khan, Vikram Singh, Avinash Patil, Sushma Wadar, et al. Attendance management system. In *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, pages 418–422. IEEE, 2015.

[78] Lawrence Sirovich and Michael Kirby. Low-dimensional procedure for the characterization of human faces. *Josa a*, 4(3):519–524, 1987.

[79] Xudong Sun, Pengcheng Wu, and Steven CH Hoi. Face detection using deep learning: An improved faster rcnn approach. *Neurocomputing*, 299:42–50, 2018.

[80] Zhan-Li Sun, De-Shuang Huang, and Yiu-Ming Cheun. Extracting nonlinear features for multispectral images by fcmc and kpca. *Digital Signal Processing*, 15(4):331–346, 2005.

[81] Zhan-Li Sun, De-Shuang Huang, Yiu-Ming Cheung, Jiming Liu, and Guang-Bin Huang. Using fcmc, fvs, and pca techniques for feature extraction of multispectral images. *IEEE Geoscience and Remote Sensing Letters*, 2(2):108–112, 2005.

[82] Melanie Swan. *Blockchain: Blueprint for a new economy*. " O'Reilly Media, Inc.", 2015.

[83] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[84] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[85] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[86] Don Tapscott and Alex Tapscott. The blockchain revolution and higher education. *Educause Review*, 52(2):11–24, 2017.

[87] Annie H Toderici and Mark Stamp. Chi-squared distance and metamorphic virus detection. *Journal of Computer Virology and Hacking Techniques*, 9(1):1–14, 2013.

[88] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.

[89] Muhamed Turkanović, Marko Hölbl, Kristjan Košič, Marjan Heričko, and Aida Kamišalić. Eductx: A blockchain-based higher education credit platform. *IEEE access*, 6:5112–5127, 2018.

[90] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. Ieee, 2001.

[91] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[92] Priyanka Wagh, Roshani Thakare, Jagruti Chaudhari, and Shweta Patil. Attendance system based on face recognition using eigen face and pca algorithms. In *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pages 303–308. IEEE, 2015.

[93] Xiang Wu, Ran He, Zhenan Sun, and Tieniu Tan. A light cnn for deep face representation with noisy labels. *IEEE Transactions on Information Forensics and Security*, 13(11):2884–2896, 2018.

[94] Cheng Xiang, XA Fan, and Tong Heng Lee. Face recognition using recursive fisher linear discriminant. *IEEE Transactions on Image Processing*, 15(8):2097–2105, 2006.

[95] Junfeng Xie, Helen Tang, Tao Huang, F Richard Yu, Renchao Xie, Jiang Liu, and Yunjie Liu. A survey of blockchain technology applied to smart cities: Research issues and challenges. *IEEE Communications Surveys & Tutorials*, 21(3):2794–2830, 2019.

[96] Yong Xu, Bob Zhang, and Zuofeng Zhong. Multiple representations and sparse representation for image classification. *Pattern Recognition Letters*, 68:9–14, 2015.

[97] Shu Zhan, Qin-Qin Tao, and Xiao-Hong Li. Face detection using representation learning. *Neurocomputing*, 187:19–26, 2016.

[98] Jun Zhang, De-Shuang Huang, Tat-Ming Lok, and Michael R Lyu. A novel adaptive sequential niche technique for multimodal function optimization. *Neurocomputing*, 69(16-18):2396–2401, 2006.

[99] Zhong-Qiu Zhao and De-Shuang Huang. A mended hybrid learning algorithm for radial basis function neural networks to improve generalization capability. *Applied Mathematical Modelling*, 31(7):1271–1281, 2007.

[100] Zhong-Qiu Zhao, De-Shuang Huang, and Wei Jia. Palmprint recognition with 2dpca+ pca based on modular neural networks. *Neurocomputing*, 71(1-3):448–454, 2007.