FaceHub: Facial Recognition Data Management in Blockchain

Akobir Ismatov, Vanessa Garza Enriquez, and Madhusudan Singh

Abstract  Real-time facial recognition systems are being widely implemented all around the world because of being efficient, reliable and very accurate systems. The Facial recognition softwares are everywhere we look and the best examples are payment methods, where we can securely make our payments without the need to enter our passwords or we can just make our groceries and the payments have done through payment methods by our identity has recognized through camera at a shop. Nowadays, Universities all around the world are also starting to use these very useful systems for taking their attendances or to authenticate their school stuff or students. This system challenge is securing the face data in database. In this chapter, we have discussed the facial recognition system (FaceHub). The requirements modelling and the real-time facial recognition system (FaceHub) that we created in order to make use of it in our University attendance system and introducing blockchain technology for facial data management, which manage the facial data in the permissioned distributed server to securely store the facial data of student. Also, we discussed the benefits of blockchain in FaceHub Data. To further, improve the security of the software by implementing the blockchain into our software.

Keywords  FaceHub · Facial image recognition · Face detection · Blockchain technology · Security

1 Introduction

The authentication systems wherever we go they are cameras everywhere and they can make our busy work process easier by doing daily routine attendance or tagging worker IDs, social media where we use them to correctly put filters to pictures or to securely enter to our applications. Most of the universities focus on student participation in class since student participation leads to effective learning and increases success rates [1]. A high participation rate in the classroom also motivates teachers...
and contributes to a more willing and informative teaching environment [2]. The most standard procedure found to influence attendance is daily attendance. Currently, there are two common ways to create attendance data available. Some teachers would like students’ names to be called and signs to be left. Other teachers would rather pass the signature sheet around. After the attendance data is collected using either one of these two methods, teachers enter the data manually into the existing system. But these non-tech methods are not efficient, because they take time and are susceptible to error and fraud [3]. All this can be solved, however, by introducing a facial recognition software for university participation to which our team will refer to as “FaceHub” (FaceHub means the collection of faces). Figure 1 has shown the system where facial recognition has managed and perform a real-time recognition, data tracking and reporting.

The data is stored in the blockchain-based cloud server and accessible from everywhere at any time only with authorized administrator. In order to make this system even more reliable and secure it’s suggested to use the permission-based blockchain server in university’s data centres so that no data manipulations or error and systems downs from the side of data cloud could happen forever. Attendance results are stored in a blockchain server database and are accessible by the professor, students and administrators.

This article presents a background study in Sect. 2, Security challenges in FaceHub in Sect. 3, Sect. 4 has discussed Blockchain Overview and Role of blockchain in FaceHub, FaceHub Requirements modelling in Sect. 5, FaceHub Methodology in Sect. 6, Implementation and Results in Sect. 7. In addition, final conclude in Sect. 8.
2 Background Study

The reason why the use of FaceHub is more reliable and cost efficient is that the current ways of attendance taking in almost all universities are out-of-date methods and software. In addition, the installation and use of other types can be very costly or unproductive. Fingerprint reading devices are very expensive to install. Moreover, a portable finger recognition device is only available to one student at a time and makes it a time-consuming process [4].

The main advantages of the proposed system are flexible usage, no equipment costs, no waste of time, easy accessibility, and more system safety, reliability and being an anti-fraud system can be achieved. We can see details in Fig. 2. To make it more clear what kind of challenges and drawbacks current attendance systems have in use, the FaceHub is going to address them. For a fixed finger recognition device at the entrance of the classroom, the teacher should attend so that students do not leave after finger recognition, which takes both the teacher and the students’ time during the course because Chintalapati It should be done under the supervision of the teacher. For RFID card lecture systems, attendance on the cards distributed to the students is available [5]. However, students can use fraudulent methods in such systems by reading the cards of their friends. In addition, an unfair absence can be registered in the system if a student forgets his or her card. The disadvantage of Bluetooth or beacon class scanning systems is that each student needs a device, which

![Advantage of facial recognition](image)

Fig. 2 Advantage of facial recognition
they would need to rely on their device’s battery for their attendance. In addition, since the Bluetooth Low Energy (BLE) system cannot determine the field limit, the attendance system may appear to students who are not currently in the classroom but who are under the Bluetooth field limit [6].

Various classroom attendance-monitoring methods are available with face recognition technology. One of these is a camera at the entrance of the classroom and face recognition of students entering to the facility [7]. However, the faces of students in this system can be recognized, students might leave the classroom afterwards and may cause errors in polling information can occur. The observation is done with a camera in the classroom, and the picture taken during the course is another method. In this case, the cameras in the system must frequently be changed to produce better images. This system is, therefore, not very helpful and can be expensive. Besides all the drawbacks, the most frequent downside is that all these methods require additional equipment. To address these disadvantages, the FaceHub app can be maintained.

3 Security Challenges

Even though the goal for our facial recognition system is authentication of the users from the trained database, this facial recognition-based authentication system should also be reliable and secure. FaceHub needs a big amount of face trained datasets and samples that will be stored in the database of it, which could lead to accessibility being taken by hackers. Hackers can also manipulate and sell the data stolen by them like the data of the users and their identity. Which is why for our software we thought of using the platform, which is secure, reliable, tamper-proof for that security breaches as well as hackers are not able to compromise the data by using the blockchain to store face images. With the blockchain technology, we could reach the absolute non-editability of a historical record. It also ensures the security of the system and makes it breach-proof. Accessibility is given by storing data on various computers for facial recognition which it will make the ledger public [8]. Blockchain will provide the security for our face recognition system with the ability of being decentralized. For example, a user wanting to edit some data for correctional purposes will need to get their changes verified by all nodes in the network. For this, the user will fail to corrupt any data in any attempt to do so. Which is why this system is highly recommended for keeping crucial data from being corrupted.

4 Blockchain Overview and Role of Blockchain in FaceHub

Blockchain technology is a very new concept, which is introduced recently, and by now, it revolutionized the security systems of newly advanced technologies [9]. For this, it is being implemented amongst all new concepts and systems where the
security is crucial and an indispensable need. To highlight the key features of the blockchain refer to Fig. 3.

We have explained the key features of blockchain technology below.

- **Decentralized Network**: The network is decentralized meaning it does not have any governing authority or a single person looking after the network. The network is controlled by the group of nodes to maintain the network making it decentralized.

- **Security**: As the need for central authority is obsolete, no one can change or modify any characteristics of the network for their benefits. Also using encryption to help the network make decisions.

- **Distributed Ledger**: All users on the system maintain the ledger on the network. This distributes the computational to ensure a better outcome.

- **Trusted Data**: Every node on the network has a copy of the digital ledger. To add new data in blockchain every node needs to verify its validity. If the majority can prove its validity, then the data can be inserted to the ledger or replaced with another. This provides the transparency of the process, which makes it tamper-proof.

### 4.1 Basic Components of Blockchain Technology

Blockchain is cryptographically connected chains of data blocks, and each of these blocks keeps track of the transactional records. All components are linked together with the help of hash function. For records to be created, a distributed network of computers shall confirm them. All the members of the network should previously take part in the previous chain creation.
4.2 Processes of Blockchain Technology

The process of the blockchain is very easy to understand. One of the nodes requests the transaction, and this transaction is broadcasted to all nodes (computers connected to the network) in the network. These nodes will receive the request and they will prove the validation of the transaction using simple math (an algorithm), and when the transaction is verified it will be written into the block and is connected to the chain.

4.3 Benefits

One of the most important advantages of the blockchain is that it is a distributed system, which will provide the availability of the data. The blocks are created in each computer in the network, so even if one of the nodes in the server or in the network fails it will not affect the workflow of the whole system, instead another node in the connection [10] will replace it. In addition, the copy of the data in blockchain is provided to everyone in the network even to the ones which are newly connected to the network which will provide the accessibly. However, having the copy of the data in each member doesn’t mean the including could be changed indeed, no changes to any copy of the data can be done without verification by whole network, and if some of the files get damaged or infected it will simply be replaced by the original copy of the data in other nodes.

4.4 Blockchain in COVID-19

Misinformation is a major concern nowadays with COVID-19-related topics due to everyone trying to give as much information as they can while at the same time providing not so accurate data making many people filled with inaccurate or false information. This has become a challenge to people who are looking to inform themselves with the right information. Which is why blockchain has become more useful for the government as it can help deal with transparent healthcare business models. Since it can protect critical patient information tamper-proof, it is now the safest way to manage information about COVID-19-related data around the world and track vaccine advancements. With a blockchain-implemented software, it is possible to keep track in real time of patients’ data, new cases, current best treatments, dangerous places to avoid, etc. This can help not only governments and civilians in search of information but also researchers and doctors who are working on ways to help end the pandemic and prevent one in the future. Currently, the World Health Organization (WHO), IBM, HACERA and Microsoft are working together to develop
a software with blockchain implementation called MiPasa to help fetch data on COVID-19 and provide critical information on virus detection [11] (Fig. 4).

4.5 Blockchain in FaceHub

The main purpose of the FaceHub is authentication of the students or other professors (professors or staff of the university), people to verify their identity or taking attendance during the class as it’s very important for university life.

The accuracy of our software is very important as it greatly affects the system, therefore, one of the main points for us was improving the accuracy as much as possible with the help of up-to-date algorithms and process of facial recognition.

In this chapter, we did not deeply dive into the security aspect, as it was not required. However, the security is a big issue for our project as the unauthorized access to the datasets or database could lead to the manipulation of the data, which may result in frauds during attendance or authentication of the users. Therefore, we found the solution in one of the current distributed technologies, which is blockchain. The blockchain is implemented in the database so the data can save in the blocks, which will make this very secure because the data stored cannot be changed until the changes are verified by all nodes in the network. While at the same time, the data becomes accessible to all the members in the server or network who must upload their images or should make changes to the image database. That is the reason why we want to implement blockchain in FaceHub to make our software more reliable and secure. Figure 5 represents the FaceHub data in blockchain networks.
5 FaceHub Requirements Modelling

5.1 Software Requirements

To make requirements more understandable, a main module flowchart as shown in Fig. 6 and use case diagram are created. The first two requirements will be explained: User requirements (defining the requirements for users with help of using more easy-to-understand terms and natural language) and System requirements (which is the complete info of the system and its functions).

5.2 User Requirements

- The FaceHub software should provide accurate recognition of the students and professors (or other staff members if included, for example, University system worker or IT department).
- The data (pictures of faces of students) should be processed correctly to identify the student.
- The FaceHub software should generate a report stating the attendance status of the students (the report shall be an excel file including attendance result of students,
Fig. 6 Flowchart of facial recognition
and it is accessible only to professor of the exact class or to the University’s system for future implementation).

- Overall time spent for recognition and attendance marking should be minimum (as the time taken for attendance is taken from the actual class time, the time for attendance done by FaceHub application should be minimum).

### 5.3 System Requirements

- Student registration to database shall be done before the software can start working.
- Attendance process for the proposed system should be most accurate including steps: Face capturing, face recognition and attendance processing.
- To open the attendance system, the lecturer of the course needs to enable the system.
- In the database, the cropped image should be converted to grayscale image and resized to a 96 × 96 pixels image and should be turned in yml file.
- If student’s data has been matched with the real-time data, the student shall be notified from the attendance system about his or her attendance being recorded (for future implementation).
- If student’s data has not been matched or has not been recorded during real-time process, the student shall be notified about his absence during class and the attendance status should be taken (for future implementation).
- The result of the attendance process should be reported to professor from attendance system (for future implementation).

### 5.4 Functional Requirements

- Users (professors and students) should be able to log in or log out from their accounts (the account must be unique for professors and students and their account privileges should differ) registered (in databases) by Universities.
- Professor should be to take attendance (including putting absence or presence to students in class stats) and he/she should be able to see the result of the attendance or the result generated by the FaceHub application.
- Professor should be able to create courses, timetables and register student to that classes/timetables she/he created.
- Students shall be able to view the results of their attendance for the class they take (they are not able to see the results of the report generated by the FaceHub application).
- Students shall be able to take their own attendance that should be verified by the class professor (only in case the professor made the error).
• Every information generated by FaceHub application (including attendance, generated report, etc.) shall be saved and kept in database for the further processing by the system.
• Application should keep the result of the attendance taken for the day in case of any errors occurs in the database or the system, to be able to fix them or to provide the required data.

5.5 Non-functional Requirements

• Adjustability: Application must be flexible to future changes and improvements (if any error occurs or disfunction in the software is found), and it should allow administrators to add new features not available in the initial release (if any need from side of university may occur in the future).
• Compatibility: The application should be able to work on both possible platforms for mobile (IOS and Android) devices.
• User friendly: The application’s user interaction interface and the application itself should be simple to understand and not complicated (as the main users of the application are non-professionals and non-familiar users, they should be able to understand how to use the application from the begging thanks to simple and understandable design of the application).
• Error-free: During the actual process of attendance by facial recognition, the application should provide the most possible accuracy for the attendance registration (99%).
• Requirement modelling: In Fig. 7 the workflow process of the application which is also provided in requirements section is explained in more understandable and clean steps with the help of Requirements model. As you can see all the functional requirements were included in the system.

6 FaceHub Implementation Method

The FaceHub software consists of two levels such as Recognition level and processing level. Figure 8 represents the requirements of FaceHub.

6.1 Recognition Level

In this level, the software is connected to the python code 1 in which the software opens the user’s camera on their device (PC or Laptop) and starts taking user’s pictures. The amount of the picture that should be taken depends on the number of requirements inside the code (for example, for our demo, we used around 250-trained
Fig. 7  Use case diagram of facial recognition requirement model

Fig. 8  FaceHub requirements modelling
pictures of each user). Then after finishing the process, the software saves the pictures into the dataset folder (which serves as a database in our HDD) that was previously stated as the path folder inside the code.

### 6.2 Sequence Diagram

In Fig. 9, we can see the sequence of action done during usage process of the FaceHub software. In this figure, User is our actor and she/he controls and initiates the whole process of recognition, which is executed and done by software.

### 6.3 Pseudocode

Algorithm: Haar Cascade Frontal Face Detection Algorithm
1. Input: Face image from camera
2. Output: Real-time image with face indicators as rectangles.
3. For $i \leftarrow 1$ to num of scales in pyramid of images do
4. Download sample image to dataset
5. Compute integral image, $\text{image}_{ii}$
6. For $j \leftarrow 1$ to num of shifts steps of sampling do
7. For $k \leftarrow 1$ to num of stages in cascade classifier do
8. For $l \leftarrow 1$ to num of filters of stage $k$ do
9. Filter detection and sample matching
10. Accumulation filter outputs
11. End for
12. If processing fails per-stage threshold then
13. Reject sample as face
14. Break this $k$ for loop
15. End if
16. End for
17. If sampling passed all per-stage checks then
18. Accept this sample as a face
19. End if
20. End for
21. End for

### 6.4 Processing Level

This level is responsible for all the processing done with help of our algorithm (in our case it is our Haar cascade), and as shown in Fig. 10 you can see the more detailed version of both processes. In this process, we also use our second and third code files. In the second file, the pictures in our database are turned into trainer file (yml file) and are ready for usage for real-time recognition process. In the third code file, we finally start our real-time recognition process with help of our previously created yml file (all the datasets that we require for real-time recognition is in that file). Face detection and recognition algorithms are performed in this layer. Precise and efficient facial detection algorithms improve the level of accuracy of facial recognition systems. The system’s failure to function will terminate the current processing and restarts if a face is not properly detected. For face detection, the methods are based on knowledge, functions, template, and statistics [12]. Viola-Jones [13] face detection method with Adaboost training is one of the best methods to use for this software because of being real-time facial recognition software [12, 14].

During the real-time recognition by user, the pictures can be matched with the previous ones in the database for the common features and similarities in shape, etc.
There are two basic classifications of face recognition based on image intensity: feature-based and appearance-based [15]. Feature-based approaches try (estimated) to represent the object such as eyes, nose, chin as there are compilations for different features. On the other hand, the appearance-based models only use the look of the object by different two-dimensional views. The algorithm based on features takes more time than appearance-based ones. The real-time attendance management system requires low-computational process time. That is why the feature-based approach is chosen for FaceHub app.

6.5 User Interface

User interface is one of the most important parts of the software, and it plays a very specific role in interaction of the user and software. User interface gives users more easy and simple control over the software [16].

As can be seen in Fig. 11 the user interface has created to be very user friendly and understandable to users. In the first image, we can see the home page which is the page for log in session for students and professors. To be able to go to the next step, users (students and professors) should enter their university id, which is given to them by university and the password that they have assigned for their accounts (for future implementation).

Features of the Home page (first image):

**Sing in:** Security option via what only authorized users by university can use the software (This option requires users ID and password to be entered in order to be able to get to their accounts with their data).
Using facial recognition: If the user does not want to enter their long password, they can simply use the facial recognition authorization function to enter to their respective accounts.

Stay logged in: Check box button which users can select in order to be able to stay logged in their sessions for the next time they enter to software (they will not be requested to enter their id and password. However, for security reasons, the stay logged in check box is active for only every 7 days after what they will be automatically deactivated and will need to be re-done).

Forgot password: In case user forgets their password, they can acquire a request from the server to renew their password (User will need to enter their email addresses connected with their accounts and given by university to prove their identity).

Features of the internal page (second image):

Log out (top left icon): The users can log out from their accounts in case they needed.

Notification (top right icon): The user is notified in case any message was sent by the server (this also includes “class attendance” started message sent to students when attendance check is started by professors and “class missed” notification to students by the server).

Take attendance of the class: This feature is only accessible to professors, and it serves to take attendance of the whole class by the professor (when pressed the phone camera starts taking real-time attendance of the class).
• **See attendance results**: With the help of this feature students and professors can see their/or taken by them attendance results.

7 Implementation and Results

We have implemented it in the following platform. Apple MacBook pro laptop, CPU = Intel core i7 3.5 GHz, RAM = 16 GB, SSD 1TB, HD 1080p Web camera, with Linux based Mac OS system all the code were done with the help of Jupiter notebook and python programming language (Fig. 12).

Hardware request is suggested. The blockchain database server is capable of handling large amounts of data, which should exceed at least 500, => GB (this database is only for keeping the data from software including face images, trained data of faces, user info, etc.). CPU Dual Core Intel Pentium (Xeon) family or higher is recommended. 16 GB of RAM or more and 4k camera is required for more accurate and quality images. Server should be capable of handling algorithms like Viola-Jones, Eigenfaces, Fisherfaces and LBP face recognition algorithms based on OpenCV. During the implementation of the software to this laptop, the execution process was quite fast due to the hardware of our laptop (however, even better results can be achieved if implemented to desktop PC with the same or less hardware capabilities). The samples (pictures) from camera have scaled to gray to be able to use via Haar Cascades. The accuracy of the facial recognition process was achieved around 80–90% which was quite accurate for the without 4K camera laptop and because of having less data samples. However, if the system has implemented to full desktop

![Fig. 12 Results of FaceHub implementation](image-url)
PC with good camera and hardware, the accuracy level around 97–99% is easily achieved.

8 Conclusion

In this article, we tried to present a real-time based facial recognition system to use it as attendance and authentication systems for university. FaceHub is a secure, user friendly, agile, time and budget efficient system. This software can make universities daily boring attendance taking routine more efficient and secure with the help of new state of the art concepts like facial recognition. With help of blockchain, we can train our security cameras to recognize people's faces and their identities. Cascades can be used to recognize features of the human face and the parts of it or the whole body in order to make the recognition process more efficient and secure. Implementing blockchain to this system will make sure of that the data kept for training or recognition process is safe. Databases protected by the blockchain can protect the whole system from the reach of possible hackers or breaches and will keep all the data safe for use and at the same time will make sure of accessibility of the system to everyone anywhere. However, universities are not the only possible users for our software, but also FaceHub can possibly be implemented in any industries and can be used anywhere like banks, big tech companies, houses, government locations or to protect important landmarks, etc.

Glossary

**FaceHub** The short name for facial recognition software (FaceHub means the collection of faces).

**Face recognition** The method of identifying or verifying the identity of an individual using their face.

**Face detection** The computer technology used in a variety of applications that identifies human faces in digital images.

**Fingerprint reading** The process of using a computer or fingerprint reader to match fingerprints against a database of known and unknown prints.

References

1. Stanca L, Econ J (2006) The effects of attendance on academic performance: panel data evidence for introductory microeconomics. Education 37(3):251–266
2. Pani PK, Kishore P (2016) Absenteeism and performance in a quantitative module: a quantile regression analysis. J Appl Res High Educ 37(3):251–266
3. Thakar U, Tiwari A, Varma S (2016) On composition of SOAP based and RESTful services. In: IEEE 6th international conference on advanced computing (IACC)
4. Basheer KPM, Raghu CV (2012) Fingerprint attendance system for classroom needs. In: Annual IEEE India conference (INDICON), pp 433–438
5. Konatham S, Chalasani BS, Kulkarni N, Taebi TE (2016) Attendance generating system using RFID and GSM. In: IEEE long island systems, applications and technology conference (LISAT)
6. Noguchi S, Niibori M, Zhou E, Kamada M (2015) Student attendance management system with Bluetooth low energy Beacon and Andorroid devices. In: 18th international conference on network based information systems, pp 710–713
7. Chintalapati S, Raghunad MV (2013) Automated attendance management system based on face recognition algorithms. In: IEEE international conference on computational intelligence and computing research
8. Saumya Sh, Jitendra M, Poonam Sh (2020) Securing face recognition system using blockchain technology. In: Bhattacharjee A et al (eds) MIND 2020, CCIS 1241. Springer Nature Singapore Pte Ltd., pp 449–460
9. Chaurasia VK, Yunus A, Singh M (2020) An overview of smart city: observation, technologies, challenges and blockchain applications. In: Singh D, Rajput N (eds) Blockchain technology for smart cities. Blockchain technologies. Springer, Singapore. https://doi.org/10.1007/978-981-15-2205-5_7
10. Singh M (2020) Blockchain technology for data management in Industry 4.0. In: Rosa Righi R, Alberti A, Singh M (eds) Blockchain technology for Industry 4.0. Blockchain technologies. Springer, Singapore. https://doi.org/10.1007/978-981-15-1137-0_3
11. Ahmed B (2020) Blockchain technology and COVID-19. OpenMind BBVA. https://www.bbvaopenmind.com/en/technology/digital-world/blockchain-technology-and-covid-19/
12. Zhu X, Ren D, Jing Z, Yan L, Lei S (2012) Comparative research of the common face detection methods. In: Second international conference on computer science and network technology, pp 1528–1533
13. Viaola P, Jones MJ (2014) Robust real-time face detection. Int J Comput Vis 57(2):137–154
14. Gupta V, Sharma D (2014) A study of various face detection methods. Int J Adv Res Comput Commun Eng 5(3):6694–6697
15. Masupha L, Zuva T, Ngwira S, Esan O (2015) Face recognition techniques, their advantages, disadvantages, and performance evaluation. In: International conference on computing, communication and security (ICCCS)
16. Saha D, Mandal A (2015) User interface design issues for easy and efficient human computer interaction: an explanatory approach. Int J Comput Sci Eng 3(1):127–135